

**Modelo de Gestão do Processo de
Venda e Desenvolvimento de
Software *On-Demand*
para MPE's**

Andrea Padovan Jubileu

Tese apresentada à Escola de Engenharia de São Carlos, da Universidade de São Paulo, como parte dos requisitos para obtenção do título de Doutor em Engenharia de Produção.

Orientador: Professor Titular Henrique Rozenfeld

São Carlos
2008

Livros Grátis

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

AUTORIZO A REPRODUÇÃO E DIVULGAÇÃO TOTAL OU PARCIAL DESTE TRABALHO, POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica preparada pela Seção de Tratamento
da Informação do Serviço de Biblioteca – EESC/USP

J91m Jubileu, Andrea Padovan
Modelo de gestão de processo de venda e desenvolvimento de software *on-demand* em MPE's / Andrea Padovan Jubileu ; orientador Henrique Rozenfeld. -- São Carlos, 2008.

Tese (Doutorado-Programa de Pós-Graduação e Área de Concentração em Engenharia de Produção -- Escola de Engenharia de São Carlos da Universidade de São Paulo, 2008.

1. Modelo de processo de software. 2. Qualidade de processo de software. 3. Melhoria de processo de software. I. Título.

FOLHA DE JULGAMENTO

Candidata: Bacharel **ANDREA PADOVAN JUBILEU**

Tese defendida e julgada em 16/12/2008 perante a Comissão Julgadora:

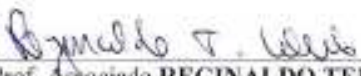

Prof. Titular **HENRIQUE ROZENFELD (Orientador)**
(Escola de Engenharia de São Carlos/USP) emovido


Prof. Dr. **DANIEL CAPALDO AMARAL**
(Escola de Engenharia de São Carlos/USP) APROVADO


Prof. Dr. **ROSELY SANCHES**
(Instituto de Ciências Matemáticas e de Computação/USP) Aprovado


Prof. Dr. **CLÊNIO FIGUEIREDO SALVIANO**
(Centro de tecnologia da Informação/CTI) Aprovado


Prof. Dr. **SANDRA CAMARGO PINTO FERRAZ FABBRI**
(Universidade Federal de São Carlos/UFSCar) Aprovado


Prof. Associado **REGINALDO TEIXEIRA COELHO**
Coordenador do Programa de Pós-Graduação em
Engenharia de Produção


Prof. Associado **GERALDO ROBERTO MARTINS DA COSTA**
Presidente da Comissão da Pós-Graduação da EESC

*Ao meu esposo Renato e
aos meus pais Geraldo e Matilde.*

AGRADECIMENTOS

A Deus, por guiar os meus passos.

Ao Renato, meu grande amor e companheiro, grande incentivador desse trabalho.

A toda minha família, em especial aos meus pais (Geraldo e Matilde) que sempre me apoiaram em todos os momentos de minha vida, aos meus irmãos (Cassiano, Adriano), minha tia Ivanilde, meu sogro (Alceu) e minha sogra (Elvira), minhas sobrinhas (Lavínia e Lorena), meu sobrinho (Igor) e minhas cunhadas (Haline e Adriane) que estiveram sempre ao meu lado e souberam compreender as minhas ausências quando necessário.

A todos que colaboraram direta ou indiretamente para que fosse possível a realização desse trabalho: Moacir, Haroldo, Robson, Douglas e, em especial, à Sayuri, Jana, Aglaê, Maria José, Marcelo (Mão), Emerson, Rogério, Marcos, Ana Cristina e Márcio.

A todos os amigos que me incentivaram durante a realização desse trabalho: Ana Lúcia, Francis, Sayuri, Jana, Simone, Ana Claudia, Helien, Eve, Eduardo, Dani, Ana Paula, Cris, Leandro, Mão, Aglaê, Maria José, Emerson, Roger, Débora, Gisele e Ivani.

A todos os professores que encontrei durante minha vida acadêmica. Em especial, à Rosely que com seu conhecimento e experiência enriqueceu minha vida profissional e pessoal, e ao prof. Henrique que me deu a oportunidade de desenvolver esse trabalho de pesquisa e, conseqüentemente, contribuiu para o meu crescimento profissional.

Aos proprietários das empresas onde foi realizada a pesquisa-ação, Paulo e Marcos, e aos funcionários dessas empresas.

Aos funcionários e docentes do Departamento de Engenharia de Produção da EESC, em especial ao Zé Luis.

RESUMO

JUBILEU, A. P. Modelo de Gestão do Processo de Venda e Desenvolvimento de Software *On-demand* para MPEs. 2008. 330p. Tese (Doutorado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2008.

A maioria das micro e pequenas empresas (MPEs) de desenvolvimento de software brasileiras são voltadas para o desenvolvimento de software *on-demand*. Normalmente, essas MPEs têm dificuldades em formalizar um processo de software padrão. O presente trabalho de pesquisa tem por objetivo apresentar uma proposta de integração de modelos/normas de capacidade de processo com modelos de ciclo de vida de software, em um contexto de gestão de processos de negócio. Como resultado, obteve-se o Modelo de Gestão do Processo de Venda e Desenvolvimento de Software *On-demand* para MPEs (ProcSoftVD - Gestão), composto pelo Método de Melhoria de Processo de Software (ProcSoftVD – Melhoria) e pelo Modelo de Processo de Venda e Desenvolvimento de Software *On-demand* para MPEs (ProcSoftVD). O Método de Melhoria de Processo de Software foi criado a partir de abordagens existentes e complementares, voltadas às MPEs. O ProcSoftVD foi originado com base no *framework Unified Process*, possibilitando a visualização do processo em duas perspectivas – fases e áreas de conhecimento, e nos modelos/normas de capacidade de processo CMMI-DEV e ISO/IEC 15504-5, elaborado em um processo iterativo e evolutivo de pesquisa-ação com a participação de duas MPEs. Um diferencial dessa proposta é a consideração de atividades de comercialização do software junto ao processo de desenvolvimento de software, o que auxilia na delimitação do escopo do projeto de desenvolvimento de software para um acordo contratual. Outro diferencial é o detalhamento das atividades do processo por meio de tarefas, sugestão de papéis desempenhados pelos responsáveis das atividades e disponibilização de *templates* com exemplos para cada um dos artefatos elaborados na execução da atividade.

Palavras-chave: Modelo de Processo de Software, Qualidade de Processo de Software, Melhoria de Processo de Software.

ABSTRACT

JUBILEU, A. P. Management Model of Selling and On-demand Software Development Process. 2008 PhD Thesis – Sao Carlos Engineering School, Sao Paulo University, Sao Carlos, 2008.

The majority of the software development micro and small companies are turned to the development of on-demand software. Usually, for these small companies the formalization of a standard process for software development is very difficult. For that reason the goal of this research is to propose an integration of capability processes models/standards within software life cycle models, in a perspective of business process management. The main result of this research is a Management Model of Selling and On-demand Software Development Process, which embraces the Software Process Improvement Method and the Selling and on-demand Software Development Process Model. The Software Process Improvement Method was build from complementary and existents tailored approaches for micro and small companies. The Selling and on-demand Software Development Process Model was created based on the models/standards of process capability CMMI-DEV and ISO/IEC 15504-5 and on the Unified Process Framework, allowing the process view from two perspectives – phases and knowledge areas. The model was elaborated in an iterative and evolutionary action-research process carried out within two micro and small companies. The originality of this proposal is the consideration of software sale activities jointly with software development process, assisting the scope delimitation of a software development project for contractual agreement. Other aspect of this research which makes it distinctive is the detail of the process activities by mean of tasks, suggestions of people roles for each activities and provision of templates with examples for each artifact created during the activities.

Key words: Software Process Model, Software Quality Process, Software Process Improvement.

Lista de Figuras/Tabelas/Quadros

Figura 2.1 – Modelo de Processo de Software Clássico (Cascata)	26
Figura 2.2 – Modelo Incremental	27
Figura 2.3 – Modelo Espiral	28
Figura 2.4 - Modelo RAD	29
Figura 2.4 – <i>Unified Process</i> . Fonte: Traduzido de (AMBLER, 2005)	31
Figura 2.5 – <i>OpenUP</i>	32
Fonte: Traduzido de (Eclipse Process Framework Community, 2008)	32
Figura 2.6 – ISO/IEC 12207 - Processos de Ciclo de Vida de Software	34
Fonte: Traduzido de (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2001)	34
Figura 3.1 – Componentes dos modelos do CMMI	53
Fonte: Traduzido de (SOFTWARE ENGINEERING INSTITUTE, 2006)	53
Figura 5.1 – Modelo de Gestão do Processo de Venda e Desenvolvimento de Software On-demand para MPEs (ProcSoftVD - GESTÃO)	74
Figura 5.2 – Primeira Versão do Modelo Genérico Do PV&DS On-demand	76
Figura 5.3 – Instância do Modelo Genérico Do PV&DS on-demand para a Empresa 1	78
Figura 5.4 - Visão Gráfica Da Fase Vender	79
Figura 5.5- Visualização gráfica do conteúdo dos artefatos do processo proposto	80
Figura 5.6 - Visualização gráfica do conteúdo do artefato “documento de requisitos”	81
Figura 5.7 - Detalhamento de uma Atividade do Modelo de Processo Proposto	82
Figura 5.8 – Exemplo de Template	83
Figura 5.9 – Meta-modelo do ProcSoftVD	86
Quadro 5.11– Associação das áreas de conhecimento do ProcSoftVD com outros modelos	91
Figura 5.12– Atividades da Área de Gestão de Conhecimento do Modelo ProcSoftVD	96
Quadro 5.13 – Quadro Comparativo dos Métodos de Melhoria de Processo	97
Tabela 5.14 – Pontuação para as áreas de conhecimento (SALVIANO, 2006)	100
Figura 5.15 – Página Principal do GProcSoftVD	105
Figura 5.16 – Exemplo de Visão Gráfica – fase de prospecção	106
Figura 5.17 – Visão Textual: Perspectiva por Fases e por Áreas de Conhecimento	107
Figura 5.18 – Exemplo de uma Atividade - fase de prospecção	108
Figura 5.19 – Exemplo de Detalhamento de Atividade	109
Figura 5.20 – Atividades da Área de Conhecimento “Gestão de Conhecimento”	110
Figura 5.21 – Mapeamento do ProcSoftVD com CMMI e ISO/IEC 15504-5	111
Figura 5.22 – Exemplo de Mapeamento da Atividade “Definir escopo do Projeto”	112
Quadro 6.1 – Características de qualidade do Modelo ProcSoftVD	120
Figura 6.2 – Gráfico de relação entre as respostas dos profissionais	120
Figura A1.1 – Planilha de Relevância (Descrição do processo)	135
Figura A1.2 – Planilha de Relevância (Comentário referente à cada questão)	136
Figura A1.3 – Planilha de Relevância (Seleção de alternativas)	136
Figura A2.1 – Tabulação da Importância versus Risco das Áreas de Conhecimento para a Empresa	137
Quadro A3.1 – Entradas, Atividades e Saídas da fase Prospecção	139
Quadro A3.2 – Entradas, Atividades e Saídas da fase Concepção	140
Quadro A3.2 – Entradas, Atividades e Saídas da fase Concepção (Cont.)	141
Quadro A3.3 – Entradas, Atividades e Saídas da fase Negociação	142
Quadro A3.3 – Entradas, Atividades e Saídas da fase Negociação (Cont.)	143
Quadro A3.4 – Entradas, Atividades e Saídas da fase Elaboração	143
Quadro A3.4 – Entradas, Atividades e Saídas da fase Elaboração (Cont.)	144
Quadro A3.5 – Entradas, Atividades e Saídas da fase Construção	144
Quadro A3.5 – Entradas, Atividades e Saídas da fase Construção (Cont.)	145
Quadro A3.6 – Entradas, Atividades e Saídas da fase Transição	145
Quadro A3.6 – Entradas, Atividades e Saídas da fase Transição (Cont.)	146
Quadro A3.7 – Entradas, Atividades e Saídas da Área de Conhecimento Comercialização	147
Quadro A3.7 – Entradas, Atividades e Saídas da Área de Conhecimento Comercialização (Cont.)	148
Quadro A3.8 – Entradas, Atividades e Saídas da Área de Conhecimento Modelagem de Negócio	149
Quadro A3.9 – Entradas, Atividades e Saídas da Área de Conhecimento Produção de Requisitos	149

Quadro A3.9 – Entradas, Atividades e Saídas da Área de Conhecimento Produção de Requisitos (Cont.).....	150
Quadro A3.10 – Entradas, Atividades e Saídas da Área de Conhecimento Projeto, Codificação & Integração de Produto.....	151
Quadro A3.11 – Entradas, Atividades e Saídas da Área de Conhecimento Verificação & Validação	152
Quadro A3.11 – Entradas, Atividades e Saídas da Área de Conhecimento Verificação & Validação (Cont.).....	153
Quadro A3.12 – Entradas, Atividades e Saídas da Área de Conhecimento Implantação.....	153
Quadro A3.13 – Entradas, Atividades e Saídas da Área de Conhecimento Aquisição.....	153
Quadro A3.14 – Entradas, Atividades e Saídas da Área de Conhecimento Medição.....	154
Quadro A3.14 – Entradas, Atividades e Saídas da Área de Conhecimento Medição (Cont.).....	155
Quadro A3.14 – Entradas, Atividades e Saídas da Área de Conhecimento Medição (Cont.).....	156
Quadro A3.15 – Entradas, Atividades e Saídas da Área de Conhecimento Garantia da Qualidade de Produto e Processo.....	156
Quadro A3.15 – Entradas, Atividades e Saídas da Área de Conhecimento Garantia da Qualidade de Produto e Processo (Cont.).....	157
Quadro A3.16 – Entradas, Atividades e Saídas da Área de Conhecimento Gestão de Requisitos...	158
Quadro A3.17 – Entradas, Atividades e Saídas da Área de Conhecimento Gestão de Mudanças e Configuração.....	158
Quadro A3.17 – Entradas, Atividades e Saídas da Área de Conhecimento Gestão de Mudanças e Configuração (Cont.).....	159
Quadro A3.18 – Entradas, Atividades e Saídas da Área de Conhecimento Gestão de Projetos.....	159
Quadro A3.18 – Entradas, Atividades e Saídas da Área de Conhecimento Gestão de Projetos (Cont.).....	160
Quadro A3.18 – Entradas, Atividades e Saídas da Área de Conhecimento Gestão de Projetos (Cont.).....	161
Quadro A3.18 – Entradas, Atividades e Saídas da Área de Conhecimento Gestão de Projetos (Cont.).....	162
Quadro A3.18 – Entradas, Atividades e Saídas da Área de Conhecimento Gestão de Projetos (Cont.).....	163
Quadro A3.18 – Entradas, Atividades e Saídas da Área de Conhecimento Gestão de Projetos (Cont.).....	164
Quadro A3.19 – Entradas, Atividades e Saídas da Área de Conhecimento Gestão de Conhecimento.....	164
Figura A3.20 – Visão Gráfica – Fase: Prospecção.....	165
Quadro A3.20.1 – Detalhamento da Atividade P01.....	165
Quadro A3.20.2 – Detalhamento da Atividade P02.....	166
Quadro A3.20.3 – Detalhamento da Atividade P03.....	167
Quadro A3.20.4 – Detalhamento da Atividade P04.....	168
Quadro A3.20.5 – Detalhamento da Atividade P05.....	169
Figura A3.21 – Visão Gráfica – Fase: Concepção.....	170
Quadro A3.21.1 – Detalhamento da Atividade Cp01.....	171
Quadro A3.21.1 – Detalhamento da Atividade Cp01 (Cont.).....	172
Quadro A3.21.2 – Detalhamento da Atividade Cp02.....	173
Quadro A3.21.3 – Detalhamento da Atividade Cp03.....	174
Quadro A3.21.4 – Detalhamento da Atividade Cp04.....	175
Quadro A3.21.5 – Detalhamento da Atividade Cp05.....	176
Quadro A3.21.6 – Detalhamento da Atividade Cp06.....	177
Quadro A3.21.7 – Detalhamento da Atividade Cp07.....	178
Quadro A3.21.8 – Detalhamento da Atividade Cp08.....	179
Quadro A3.21.9 – Detalhamento da Atividade Cp09.....	180
Quadro A3.21.10 – Detalhamento da Atividade Cp10.....	181
Quadro A3.21.11 – Detalhamento da Atividade Cp11.....	182
Quadro A3.21.12 – Detalhamento da Atividade Cp12.....	183
Quadro A3.21.13 – Detalhamento da Atividade Cp13.....	183
Quadro A3.21.14 – Detalhamento da Atividade Cp14.....	184
Figura A3.22 – Visão Gráfica – Fase: Negociação.....	185
Quadro A3.22.1 – Detalhamento da Atividade N01.....	186
Quadro A3.22.2 – Detalhamento da Atividade N02.....	187
Quadro A3.22.3 – Detalhamento da Atividade N03.....	187

Quadro A3.22.4 – Detalhamento da Atividade N04	188
Quadro A3.22.5 – Detalhamento da Atividade N05	188
Quadro A3.22.6 – Detalhamento da Atividade N06	189
Quadro A3.22.7 – Detalhamento da Atividade N07	190
Quadro A3.22.8 – Detalhamento da Atividade N08	190
Quadro A3.22.9 – Detalhamento da Atividade N09	191
Figura A3.23 – Visão Gráfica – Fase: Elaboração	192
Quadro A3.23.1 – Detalhamento da Atividade E01	193
Quadro A3.23.2 – Detalhamento da Atividade E02	194
Quadro A3.23.3 – Detalhamento da Atividade E03	195
Quadro A3.23.4 – Detalhamento da Atividade E04	196
Quadro A3.23.5 – Detalhamento da Atividade E05	196
Quadro A3.23.5 – Detalhamento da Atividade E05 (Cont.)	197
Quadro A3.23.6 – Detalhamento da Atividade E06	198
Quadro A3.23.7 – Detalhamento da Atividade E07	199
Quadro A3.23.8 – Detalhamento da Atividade E08	199
Figura A3.24 – Visão Gráfica – Fase: Construção	200
Quadro A3.24.1 – Detalhamento da Atividade Ct01	201
Quadro A3.24.2 – Detalhamento da Atividade Ct02	201
Quadro A3.24.2 – Detalhamento da Atividade Ct02 (Cont.)	202
Quadro A3.24.3 – Detalhamento da Atividade Ct03	202
Quadro A3.24.4 – Detalhamento da Atividade Ct04	203
Quadro A3.24.5 – Detalhamento da Atividade Ct05	203
Quadro A3.24.5 – Detalhamento da Atividade Ct05 (Cont.)	204
Quadro A3.24.6 – Detalhamento da Atividade Ct06	204
Quadro A3.24.7 – Detalhamento da Atividade Ct07	205
Figura A3.25 – Visão Gráfica – Fase: Transição	206
Quadro A3.25.1 – Detalhamento da Atividade T01	207
Quadro A3.25.2 – Detalhamento da Atividade T02	207
Quadro A3.25.3 – Detalhamento da Atividade T03	208
Quadro A3.25.4 – Detalhamento da Atividade T04	209
Quadro A3.25.5 – Detalhamento da Atividade T05	210
Quadro A3.25.6 – Detalhamento da Atividade T06	211
Quadro A3.25.7 – Detalhamento da Atividade T07	212
Quadro A5.1 – Legenda do Mapeamento das Atividades	293
Quadro A5.1 – Legenda do Mapeamento das Atividades (Cont.)	294
Quadro A5.2 – Mapeamento das Atividades da Fase de Prospecção	294
Quadro A5.3 – Mapeamento das Atividades da Fase de Concepção	295
Quadro A5.3 – Mapeamento das Atividades da Fase de Concepção (Cont.)	296
Quadro A5.3 – Mapeamento das Atividades da Fase de Concepção (Cont.)	297
Quadro A5.3 – Mapeamento das Atividades da Fase de Concepção (Cont.)	298
Quadro A5.3 – Mapeamento das Atividades da Fase de Concepção (Cont.)	299
Quadro A5.4 – Mapeamento das Atividades da Fase de Negociação	299
Quadro A5.4 – Mapeamento das Atividades da Fase de Negociação (Cont.)	300
Quadro A5.4 – Mapeamento das Atividades da Fase de Negociação (Cont.)	301
Quadro A5.5 – Mapeamento das Atividades da Fase de Elaboração	302
Quadro A5.5 – Mapeamento das Atividades da Fase de Elaboração (Cont.)	303
Quadro A5.5 – Mapeamento das Atividades da Fase de Elaboração (Cont.)	304
Quadro A5.5 – Mapeamento das Atividades da Fase de Elaboração (Cont.)	305
Quadro A5.6 – Mapeamento das Atividades da Fase de Construção	306
Quadro A5.6 – Mapeamento das Atividades da Fase de Construção (Cont.)	307
Quadro A5.6 – Mapeamento das Atividades da Fase de Construção (Cont.)	308
Quadro A5.7 – Mapeamento das Atividades da Fase de Transição	309
Quadro A5.7 – Mapeamento das Atividades da Fase de Transição (Cont.)	310
Quadro A5.7 – Mapeamento das Atividades da Fase de Transição (Cont.)	311
Quadro A8.1 – Transcrição das Respostas – Área de Conhecimento Produção de Requisitos	321
Quadro A8.2 – Transcrição das Respostas – Área de Conhecimento Modelagem de Negócios	321
Quadro A8.3 – Transcrição das Respostas – Área de Conhecimento Comercialização	322
Quadro A8.4 – Transcrição das Respostas – Área de Conhecimento	322
Projeto, Codificação & Integração de Produto	322

Quadro A8.5 – Transcrição das Respostas – Área de Conhecimento Aquisição	323
Quadro A8.6 – Transcrição das Respostas – Área de Conhecimento Medição	323
Quadro A8.7 – Transcrição das Respostas – Área de Conhecimento Implantação	323
Quadro A8.8 – Transcrição das Respostas – Área de Conhecimento V&V	324
Quadro A8.9 – Transcrição das Respostas – Área de Conhecimento Garantia da Qualidade de Produto e Processo	324
Quadro A8.10 – Transcrição das Respostas – Área de Conhecimento Gestão de Conhecimento ...	324
Quadro A8.11 – Transcrição das Respostas – Área de Conhecimento Gestão de Requisitos	325
Quadro A8.12 – Transcrição das Respostas – Área de Conhecimento Gestão de Projetos	325
Quadro A8.13 – Transcrição das Respostas – Área de Conhecimento Gestão de Mudanças e Configuração	326

Lista de Abreviaturas e Siglas

ASPE/MSC	<i>Approach for Software Process Establishment in Micro and Small Companies</i>
BPM	<i>Business Process Management</i>
BPR	<i>Business Process Reengineering</i>
CMMI	<i>Capability Maturity Model Integration</i>
CMMI-DEV	<i>CMMI for Development</i>
IDEAL	<i>Initiating, Diagnosing, Establishing, Acting, Learning</i>
ISO/IEC 12207	<i>Information technology -- Software life cycle processes</i>
ISO/IEC 15504	<i>Information technology -- Process assessment</i>
ISO/IEC 15504-5	<i>Information technology -- Process assessment - Part 5: An exemplar Process Assessment Model</i>
KM	<i>Knowledge Management</i>
MPEs	<i>Micro e Pequenas Empresas</i>
MPS	<i>Melhoria do Processo de Software</i>
MPS.BR	<i>Melhoria de Processos do Software Brasileiro</i>
MSC	<i>Micro and Small Companies</i>
PA	<i>Process Area (CMMI)</i>
PDCA	<i>Plan, Do, Check, Action</i>
PDS	<i>Processo de Desenvolvimento de Software</i>
ProcSoftVD	<i>Modelo do Processo de Venda e Desenvolvimento de Software On-demand para MPEs</i>
ProcSoftVD - Gestão	<i>Modelo de Gestão do Processo de Venda e Desenvolvimento de Software On-demand para MPEs</i>
ProcSoftVD - Melhoria	<i>Método de Melhoria do Processo de Venda e Desenvolvimento de Software On-demand para MPEs</i>
PRO2PI	<i>Perfil de Capacidade de Processo para Melhoria de Processo</i>
PRO2PI-WORK	<i>PRO2PI Establishment Workshop Method</i>
PSEE	<i>Process-Centered Software Engineering Environment</i>
PV&DS	<i>Processo de Venda e Desenvolvimento de Software</i>
RAD	<i>Rapid Application Development</i>
RUP	<i>Rational Unified Process</i>
SG	<i>Specific Goal</i>
SOFTEX	<i>Associação para Promoção da Excelência do Software Brasileiro</i>
SPI	<i>Software Process Improvement</i>
TI	<i>Tecnologia da Informação</i>
TQM	<i>Total Quality Management</i>
UP	<i>Unified Process</i>

SUMÁRIO

LISTA DE FIGURAS/TABELAS/QUADROS.....	8
LISTA DE ABREVIATURAS E SIGLAS	12
1. INTRODUÇÃO.....	15
1.1. CONTEXTO DO TRABALHO DE PESQUISA.....	15
1.2. QUESTÃO DE PESQUISA	18
1.3. OBJETIVO.....	18
1.4. JUSTIFICATIVA	18
1.5. LIMITAÇÕES	21
1.6. ESTRUTURA DE APRESENTAÇÃO DO TRABALHO	22
2. PROCESSO DE SOFTWARE.....	24
2.1. CONSIDERAÇÕES INICIAIS	24
2.2. MODELO CASCATA	26
2.3. MODELO INCREMENTAL.....	26
2.4. MODELO ESPIRAL	28
2.5. RAD – <i>RAPID APPLICATION DEVELOPMENT</i>	29
2.6. UNIFIED PROCESS	30
2.7. OPEN/UP.....	32
2.8. ISO/IEC 12207 – PROCESSOS DE CICLO DE VIDA DE SOFTWARE.....	33
2.9. CONSIDERAÇÕES FINAIS	34
3. MELHORIA DE PROCESSO DE SOFTWARE.....	37
3.1. CONSIDERAÇÕES INICIAIS	37
3.2. GESTÃO DE PROCESSOS DE NEGÓCIO – CONCEITOS BÁSICOS	38
3.3. ALGUMAS ABORDAGENS PARA MELHORIA DE PROCESSO.....	41
3.3.1. PDCA.....	41
3.3.2. IDEAL.....	41
3.3.3. <i>Ciclo de Melhoria da ISO/IEC 15504</i>	42
3.3.4. <i>Metodologia de Gestão de Mudanças</i>	43
3.3.5. <i>PRO2PI-CYCLE</i>	45
3.3.6. <i>ASPE/MSA</i>	47
3.4. MODELOS DE REFERÊNCIA UTILIZADOS PARA MELHORIA DE PROCESSO	48
3.4.1. <i>Framework ISO/IEC 15504 e Modelo 15504-5</i>	48
3.4.2. <i>Framework SEI CMMI e Modelo CMMI-DEV</i>	52
3.4.3. <i>MPS.BR e Modelo MR-MPS</i>	61
3.4.4. <i>ISO 9000</i>	62
3.4.5. <i>PMBOK</i>	63
3.4.6. <i>SWEBOK</i>	64
3.5. CONSIDERAÇÕES FINAIS	64
4. METODOLOGIA DE PESQUISA.....	65
4.1. CONSIDERAÇÕES INICIAIS	65
4.2. ABORDAGENS E MÉTODOS DE PESQUISA.....	66
4.3. ETAPAS DO TRABALHO	68
4.4. CONSIDERAÇÕES FINAIS	73
5. MODELO DE GESTÃO DO PROCESSO DE VENDA E DESENVOLVIMENTO DE SOFTWARE ON-DEMAND PARA MPES (PROCSoftVD – GESTÃO).....	74
5.1. CONSIDERAÇÕES INICIAIS	74
5.2. EVOLUÇÃO DO MODELO DE PROCESSO DE VENDA E DESENVOLVIMENTO DE SOFTWARE <i>ON-DEMAND</i> PARA MPES (PROCSoftVD).....	75
5.2.1. <i>Requisitos do ProcSoftVD</i>	85
5.2.2. <i>Meta-modelo do ProcSoftVD</i>	86
5.2.3. <i>Perspectivas do ProcSoftVD</i>	87

5.3.	MÉTODO DE MELHORIA DO PROCESSO DE VENDA E DESENVOLVIMENTO DE SOFTWARE <i>ON-DEMAND</i> PARA MPES (PROCSOFTVD – MELHORIA)	97
5.4.	PUBLICAÇÃO WEB DO MODELO DE GESTÃO DO PROCESSO DE VENDA E DESENVOLVIMENTO <i>ON-DEMAND</i> PARA MPES	104
5.5.	CONSIDERAÇÕES FINAIS	112
6.	APLICAÇÕES E RESULTADOS DO MODELO PROCSOFTVD - GESTÃO.....	113
6.1.	CONSIDERAÇÕES INICIAIS	113
6.2.	APLICAÇÃO DO PROCSOFTVD - GESTÃO	113
6.3.	ANÁLISE DOS PONTOS FORTES E FRACOS DO MODELO PROCSOFTVD	114
6.4.	ANÁLISE DAS CARACTERÍSTICAS DE QUALIDADE DO MODELO PROCSOFTVD.....	119
6.5.	CONSIDERAÇÕES FINAIS	122
7.	CONCLUSÕES E TRABALHOS FUTUROS.....	123
7.1.	CONSIDERAÇÕES INICIAIS	123
7.2.	CONCLUSÕES.....	123
7.3.	TRABALHOS FUTUROS.....	125
7.4.	CONSIDERAÇÕES FINAIS	125
	REFERÊNCIAS.....	126
	APÊNDICE 1 – QUESTIONÁRIO PARA LEVANTAMENTO DE ÁREAS DE CONHECIMENTO RELEVANTES PARA ALGUMAS MPES.....	132
	APÊNDICE 2 – RESULTADO DA ANÁLISE DO QUESTIONÁRIO DEFINIDO NO APÊNDICE 1....	137
	APÊNDICE 3 – MODELO PROCSOFTVD (ATIVIDADES)	139
	APÊNDICE 4 – MODELO PROCSOFTVD (<i>TEMPLATES</i>).....	213
	APÊNDICE 5 – MODELO PROCSOFTVD (MAPEAMENTO COM O CMMI E ISO/IEC 15504).....	293
	APÊNDICE 6 – APLICAÇÃO DO PROCSOFTVD – MELHORIA.....	312
	APÊNDICE 7 – QUESTIONÁRIO DE AVALIAÇÃO DO PROCSOFTVD.....	317
	APÊNDICE 8 – PONTOS FORTES E FRACOS TRANSCRITOS DAS RESPOSTAS DO QUESTIONÁRIO ENTREGUE AOS PROFISSIONAIS	321
	APÊNDICE 9 – RESPOSTAS DO QUESTIONÁRIO RESPONDIDO PELOS PROFISSIONAIS.....	327

1. INTRODUÇÃO

1.1. Contexto do Trabalho de Pesquisa

Segundo uma pesquisa realizada pelo Ministério da Ciência e Tecnologia (MCT, 2005) sobre população alvo constituída por empresas desenvolvedoras de software (de pacote, sob encomenda (*on-demand*), embarcado e para uso próprio) e por empresas distribuidoras ou editoras de software de terceiros, no total de 488 empresas, 396 são empresas que desenvolvem software *on-demand* (81,1%). Adotando-se o critério aplicado à força de trabalho efetiva - microempresas (de 1 a 9 pessoas), pequenas (de 10 a 49 pessoas), médias (de 50 a 99 pessoas) e grandes (100 ou mais pessoas) - 80,9% das micro empresas e 85,9% das pequenas empresas desenvolvem software *on-demand*, confirmando-se a predominância de micro e pequenas empresas (MPEs) no cenário de desenvolvimento de software *on-demand* no Brasil.

Em busca de melhorar tanto a qualidade de seus produtos de software quanto a produtividade do desenvolvimento de software, cada vez mais as MPEs estão percebendo a importância da existência de um processo¹ de software² formalizado na empresa e investindo na melhoria do processo de software. Essa afirmação é pautada nos resultados de uma revisão sistemática, elaborada por PINO et al. (2008), realizada nas fontes: Science Direct, Wiley InterScience, IEEE Digital Library, ACM Digital Library e um relatório especial “Proceedings of the First International Research Workshop for Process Improvement in Small Settings” da SEI (*Software Engineering Institute*). No contexto brasileiro, comprova-se um aumento do desempenho de processos de software o que auxilia na melhoria da qualidade de produtos de software e no aumento das vantagens competitivas das empresas brasileiras tanto no mercado nacional quanto internacional (ROCHA et al., 2005).

¹ Processo: segundo a ABNT (Associação Brasileira de Normas Técnicas), um conjunto de atividades inter-relacionadas ou interativas, que transforma insumos (entradas) em produtos (saídas).

² Processo de Software: segundo a IEEE (*Institute of Electrical and Electronics Engineers*), trata-se de uma abordagem sistemática, disciplinada e quantificável para o desenvolvimento, operação e manutenção do software.

Um processo de software bem estabelecido, compreendido e controlado auxilia aos desenvolvedores obter um produto final com qualidade (SOFTWARE ENGINEERING INSTITUTE, 2006).

Para o estabelecimento de um processo de desenvolvimento de software podem ser usados como referências: modelos³ de processo⁴ que descrevem atividades, como o Modelo Cascata, Incremental, Espiral e RAD (*Rapid Application Development*) (PRESSMAN, 2006; SOMMERVILLE, 2007); um *framework* de processo, tal como o UP (*Unified Process*) (JACOBSON et al., 1999); uma norma, como a ISO/IEC 12207 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2001), que define as atividades, objetivos e resultados de processos relacionados ao ciclo de vida do software e/ou, ainda, modelos e normas de capacidade de processo⁵ que prescrevem melhores práticas, tais como CMMI (SOFTWARE ENGINEERING INSTITUTE, 2002), ISO/IEC 15504 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2003) e MPS.BR (WEBER et al., 2005a), metodologias ágeis, como *Extreme Programming* (WELLS, 2006) entre outras referências.

De acordo com os dados disponibilizados pelo PBQP Software⁶ (WEBER et al., 2006): 63,3% conhecem, mas não usam o CMMI (SOFTWARE ENGINEERING INSTITUTE, *op. cit*); 65% conhecem, mas não usam a norma ISO/IEC 12207 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, *op. cit*) e 71,1% conhecem, mas não usam a norma ISO/IEC 15504 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, *op. cit*).

³ Segundo Vernadat (1996), um modelo é uma representação útil de algum objeto. É uma abstração da realidade expressa em termos de algum formalismo (ou linguagem) definido por um método de modelagem em função do objetivo do usuário.

⁴ Modelos de processo de software: podem ser prescritivos, quando enfatizam a definição, identificação e aplicação detalhada de atividades e tarefas de processo, e podem ser ágeis, quando enfatizam a adaptabilidade (PRESSMAN, 2006). Neste trabalho de pesquisa, são chamados “modelos de processo de software” aqueles a partir dos quais se pode definir, identificar e aplicar detalhadamente atividades e tarefas de processo. Já os de caráter ágil são denominados “metodologias ágeis” e não modelos.

⁵ Modelos/Normas de capacidade de processo: segundo a ISO/IEC 15504, capacidade de processo é uma caracterização da habilidade do processo atingir aos objetivos de negócio atuais ou futuros. Sendo assim, neste trabalho de pesquisa são denominados “modelos/normas de capacidade de processo” aqueles que possibilitam a determinação da capacidade de um determinado processo.

⁶ Programa Brasileiro da Qualidade e Produtividade em Software que tem por objetivo atingir padrões internacionais de Qualidade e Produtividade no Setor de Software no Brasil. O PBQP Software é composto por voluntários, interessados na melhoria da qualidade e produtividade em software, ligados ao Governo, Academia e Indústria.

Devido à dificuldade e alto custo de implantação e avaliação de modelos, como o CMMI, pelas MPEs e a necessidade de expandir o mercado de software brasileiro, a Associação para Promoção da Excelência do Software Brasileiro (SOFTEX) iniciou um programa mobilizador denominado MPS.BR - Melhoria de Processo de Software Brasileiro (WEBER et al., 2005a) - com o intuito de melhorar continuamente a qualidade de software no Brasil a um custo acessível (WEBER & ARAUJO, 2006; ROCHA et al., 2006).

O Modelo MPS.BR (Melhoria de Processos do Software Brasileiro) teve como bases de construção as normas ISO/IEC 12207 e a ISO/IEC 15504 e as melhores práticas da engenharia de software, além de manter compatibilidade com o CMMI. A sua originalidade não está no conteúdo, mas sim na estratégia de implementação, criada para a realidade brasileira (SOFTEX, 2007).

Apesar da estratégia de implementação do MPS.BR ser mais adequada às MPEs brasileiras, os relatos de implantação/melhoria de processos (em vários níveis de maturidade estabelecidos pelo MPS) por parte das MPEs citam o auxílio de consultores e parcerias com universidades (BORSSATTO, 2008; MONTEIRO et al., 2008; OLIVEIRA et al., 2008; PORTO et al., 2008; SCHEID et al., 2008; VARGAS, et al., 2008). Este fato evidencia certa dificuldade por parte dessas MPEs em definir, implantar e melhorar seus processos.

Melhorar continuamente os processos de negócio é um dos objetivos da Gestão de Processos de Negócio (BPM) - uma fonte de vantagem competitiva para a empresa (HUNG, 2006). Segundo Jeston & Nelis (2008) BPM tem por princípio alcançar os objetivos estratégicos da organização por meio da melhoria, gestão e controle de processos de negócio essenciais.

Considerando esse contexto, seria bastante útil alguma maneira de permitir às próprias MPEs definirem, implantarem e melhorarem seu processo de software, atendendo a modelos de capacidade de processo tais como o CMMI e a ISO/IEC 15504-5, de modo a minimizar a necessidade de consultorias externas.

1.2. Questão de Pesquisa

De acordo com o exposto anteriormente, foi originada a seguinte questão de pesquisa - a cuja resposta se pretende contribuir:

“Considerando o contexto de gestão de processos de negócio, é possível criar um modelo que auxilie as MPEs a definir seu processo padrão⁷ de venda e desenvolvimento de software on-demand, minimizando suas dificuldades em utilizar modelos/normas de capacidade de processo, tais como CMMI, ISO/IEC 15504-5 e MPS.BR, e minimizando a necessidade de consultorias externas?”

1.3. Objetivo

Considerando a questão de pesquisa, este trabalho tem como principal objetivo criar um modelo que oriente uma MPE, de maneira mais detalhada do que a sugerida pelos modelos de capacidade de processo, na definição de seu processo padrão de venda e desenvolvimento de software *on-demand*, considerando o contexto de melhoria de processo.

O objetivo secundário do trabalho é possibilitar que o modelo a ser criado permita que a empresa selecione as áreas de conhecimento que serão abordadas em cada ciclo de melhoria do processo, de acordo com as suas prioridades.

1.4. Justificativa

Em busca de melhorar a qualidade de seus produtos de software e de ter maior visibilidade das atividades do projeto de desenvolvimento de software, cada vez mais as empresas estão enxergando a importância de melhorar o seu processo de software e ter um

⁷ Processo padrão: segundo o estabelecido no CMMI, é uma definição operacional dos processos básicos que guiam o estabelecimento de um processo comum em uma organização. Um processo padrão descreve os elementos de processo fundamentais que se espera serem incorporados em algum processo definido (processo adaptado a partir do conjunto de processo padrão, de acordo com as diretrizes de adaptação da organização).

processo formalizado que sirva de guia para todos os envolvidos em um projeto, permitindo uma visão comum entre eles.

Para auxiliar nessa missão, há na literatura, os chamados modelos de processo de software, *frameworks* e normas de processo de software que têm por objetivo orientar na definição de um processo de software e, também, os métodos, modelos e abordagens para orientarem na melhoria do processo de software.

Quanto aos métodos, modelos e abordagens que orientam na melhoria do processo de software, há vários na literatura, como Ciclo PDCA (DEMING, 1986), Abordagem IDEALSM (MCFEELEY, 1996), Ciclo de Melhoria da ISO/IEC 15504 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2003), Metodologia de Mudanças (COSTA, 2006) e alguns voltados para MPEs por serem mais detalhados, como PRO2PI-CYCLE (SALVIANO, 2006) e ASPE/MS (WEBER et al., 2005b).

Quanto aos modelos de processo de software, normalmente, eles prescrevem apenas as atividades a serem realizadas sem fornecer detalhes, que facilitariam o entendimento e uso do modelo por parte das MPEs. Alguns exemplos desses detalhes são: os responsáveis pelas atividades, suas tarefas, os artefatos de entrada necessários para a realização das atividades, os artefatos resultantes da execução das atividades, recursos que podem ser utilizados e *templates* que fornecem informações do conteúdo de cada um dos artefatos.

A maioria dos modelos de processo de software existentes na literatura foca o desenvolvimento do software. Entretanto, neste trabalho de pesquisa considera-se que a comercialização do software é uma área relevante, no caso de desenvolvimento *on-demand* (sob encomenda), pois a elicitação dos requisitos do produto a ser desenvolvido é inicialmente realizada durante as atividades de comercialização, a fim de se ter subsídio para negociar um contrato. Assim, essa área deve ser considerada juntamente ao desenvolvimento de software.

Algumas limitações dos modelos de processo de software existentes na literatura são: não abordar a área de comercialização junto ao desenvolvimento de software *on-*

demand e não dar ênfase nas atividades de gestão de conhecimento e gestão de habilidades e competência em meio ao desenvolvimento do software. Por isso, como meta para o modelo proposto buscou-se incorporar as atividades relacionadas às áreas de comercialização, gestão de conhecimento e gestão de habilidades e competências.

Existem processos, tal como o RUP (*Rational Unified Process*) - embutido em algumas ferramentas da Rational (KRUTCHEN, 2004) e o *OpenUP* (ECLIPSE PROCESS FRAMEWORK COMMUNITY, 2008) – um processo unificado ágil embutido na ferramenta *free EPF Composer* (*Eclipse Process Framework*) - que atendem às necessidades anteriormente descritas e poderiam servir como modelo a ser utilizado pelas MPEs, mas também não incluem atividades relacionadas à comercialização de software, o que auxiliaria as MPEs durante a negociação da venda do software para o cliente e acompanhamento do fluxo de caixa e, também, não incluem atividades relacionadas ao gerenciamento da memória organizacional da empresa - Gestão de Conhecimento - e nem atividades relacionadas à gestão de habilidades e competências dentro da empresa o que auxilia a criar a sua memória organizacional.

A engenharia de software pode ser vista como um processo de intensivo conhecimento (BOGEN et al., 2005). Isso porque ela consiste de diferentes campos de conhecimento, tais como análise de requisitos, projeto de software, teste de software e gestão de configuração de software (SWEBOK, 2004), e todos esses campos de conhecimento têm em comum a existência de uma grande variedade de fontes, uma alta demanda por comunicação, um período de vida curto de conhecimento e alto custo no processo.

É importante para uma organização de software saber trilhar quem conhece “o quê” para fazer uso do conhecimento não documentado (conhecimento tácito). Uma boa solução para esse problema é a gestão de competências ou gestão de habilidades. O aprendizado a partir da experiência requer uma memória de projeto e produto. O ambiente no qual o engenheiro de software conduz seu trabalho diário dá suporte à criação da memória. O controle de versão, gestão de mudanças, documentação de decisões de projeto (*design*

rationale) e rastreabilidade de requisitos são exemplos de práticas de engenharia de software que auxiliam a construir memórias como uma forma efetiva de desenvolvimento de software (RUS & LINDVALL, 2002).

Apesar de algumas barreiras para o uso efetivo de gestão de conhecimento (KM – *Knowledge Management*) na engenharia de software (DeSOUZA, 2003), foram encontradas na literatura várias abordagens de KM no PDS que relatam sobre experiências bem sucedidas (MONTONI et al., 2004a, 2004b, 2005 e 2007; FALBO et al., 2004 e SEGRINI et al., 2006; LIMA et al., 2006). Um exemplo de barreira é a resistência de ser conhecido como um especialista: alguns especialistas têm receio de serem alocados somente em projetos relacionados às suas experiências no passado ao invés de serem lançados a ele desafios intelectuais (oportunidade de aprendizado) (DeSOUZA, 2003).

Pode-se dizer que, um processo de software também deve considerar, além das atividades de desenvolvimento propriamente ditas, outras atividades relacionadas aos processos organizacionais e de apoio, tais como gestão de projeto, gestão de conhecimento, gestão de requisitos e gestão de mudanças e de configuração. Essas atividades são abordadas em modelos e normas, tais como ISO/IEC 12207, CMMI, ISO/IEC 15504 e MPS.BR, que prescrevem melhores práticas com o intuito de auxiliar na definição e melhoria de processos de software. Contudo, para as MPEs, definir seu processo padrão a partir desses modelos é uma tarefa árdua, pois requer especialistas com conhecimento em qualidade de software e em modelagem de processo.

1.5. Limitações

O modelo proposto neste trabalho de pesquisa tem as seguintes limitações:

- Não está inserido em um Ambiente de Engenharia de Software Centrado em Processo (PSEE), o que facilitaria para as MPEs a adequação do modelo para a sua realidade. Um PSEE (*Process-Centered Software Engineering Environment*) também é conhecido por PCE (*Process Centered Environments*), PSE (*Process*

Sensitive Environments) ou PSS (*Process Support System*) (ARBAQUI et al., 2002). PSEE constitui um tipo especial de ambiente de desenvolvimento de software para apoiar a definição de processos de software, objetivando automatizar a gerência do desenvolvimento. Esses ambientes provêm serviços para análise, simulação, execução e reutilização das definições de processos, que cooperam no aperfeiçoamento contínuo de processos.

- Atende somente às áreas de processo definidas para se alcançar o nível 2 de maturidade do CMMI e às áreas de processo da categoria Engenharia para se alcançar o nível 3 de maturidade. Sendo assim, processos como suporte e manutenção do software não são abordados pelo modelo elaborado neste trabalho de pesquisa.
- Não está mapeado diretamente com os *resultados esperados* definidos pelo MPS.BR. Entretanto, uma vez que o MPS.BR foi elaborado em consonância com os modelos internacionais CMMI e ISO/IEC 15504-5 e o modelo elaborado neste trabalho de pesquisa está mapeado com os *objetivos específicos* do CMMI-DEV e com as *práticas-base* da ISO/IEC 15504-5, pode-se dizer que o modelo elaborado atende aos *resultados esperados* do MPS.BR.
- Não atende aos princípios definidos por metodologias ágeis, como *Extreme Programming* (XP), SCRUM, FDD, entre outras.
- Para ser utilizado como meio de certificação da empresa em nível 2 de maturidade CMMI-DEV, toda vez que o CMMI-DEV for atualizado, o mapeamento das atividades do ProcSoftVD em relação ao CMMI e, se necessário, as atividades do ProcSoftVD devem ser atualizadas.

1.6. Estrutura de Apresentação do Trabalho

Este trabalho está organizado em capítulos. No Capítulo 2 são apresentados alguns modelos de processo de software que auxiliam as empresas a definirem os processos e

atividades que farão parte do seu processo padrão e a norma ISO/IEC 12207 que define os processos de ciclo de vida do software.

O Capítulo 3 apresenta alguns conceitos básicos relacionados à gestão de processos de negócio (que tem como um dos objetivos a melhoria dos processos) e algumas abordagens, métodos e modelos que podem ser utilizados na melhoria de processo de software. Apresenta, também, alguns modelos de capacidade de processo utilizados como referência pelas empresas para definirem o seu processo de software padrão e os outros modelos que poderiam ser utilizados como referência.

No Capítulo 4 são apresentados as abordagens e métodos de pesquisa, e as etapas realizadas no trabalho.

O Capítulo 5 apresenta o Modelo de Gestão do Processo de Venda e Desenvolvimento de Software *On-Demand* em MPE's (ProcSoftVD - Gestão), resultante desse trabalho de pesquisa.

No Capítulo 6 são apresentadas as aplicações e análises realizadas no ProcSoftVD - Gestão, além dos resultados das aplicações. E, finalmente, no Capítulo 7 são apresentados as conclusões e trabalhos futuros, as referências bibliográficas e os apêndices.

2. PROCESSO DE SOFTWARE

2.1. Considerações Iniciais

Segundo a IEEE, a engenharia de software é a aplicação de uma abordagem sistemática, disciplinada e quantificável para o desenvolvimento, operação e manutenção do software. Sistemática por partir do princípio de que existe um processo de desenvolvimento definindo as atividades que deverão ser executadas. Disciplinada, pois parte do princípio de que os processos definidos serão seguidos. Quantificável por que se deve definir um conjunto de medidas a serem extraídas do processo durante o desenvolvimento, de forma que as tomadas de decisão relacionadas ao desenvolvimento do software (por exemplo, melhoria de processo) sejam embasadas em dados reais (SPÍNOLA & ÁVILA, 2008). Essa abordagem refere-se a um processo de software.

Segundo Paulk et al. (1995), para um processo de software funcionar satisfatoriamente, deve possuir: procedimentos e métodos que descrevam a relação entre as tarefas; ferramentas e equipamentos que dêem suporte à realização das tarefas, simplificando e automatizando o trabalho; pessoas com perfil adequado, treinadas nos métodos e nas ferramentas para poderem realizar as atividades adequadamente.

Para Salviano (2006), *“processo de software é o que as pessoas fazem para um determinado propósito, utilizando suas habilidades e conhecimento, com o apoio de artefatos, ferramentas e outros recursos, para produzir software e seus produtos associados”*.

Devido à alta rotatividade de pessoas na indústria de software, há uma grande probabilidade dos desenvolvedores originais não estarem disponíveis quando surgirem os problemas e forem necessárias modificações (FIGUEIREDO et al., 2006). Esse é um dos motivos pelo qual se justifica a importância de utilizar processos como guias

de condução das atividades fundamentais de processo de desenvolvimento de software (PDS) e das atividades gerenciais, organizacionais e de apoio relacionados ao PDS.

Há na literatura vários modelos de processo de software que surgiram na tentativa de auxiliar na definição de processos de software com qualidade – cada um deles com suas peculiaridades (PRESSMAN, 2006). Modelos de processo de software são abstrações do processo de software descritos com uma linguagem formal ou semi-formal denominada PML (*Process Modeling Language or Formalism*) (CONRADI et al., 1992⁸ apud ARBAQUI et al., 2002; LONCHAMP, 1993⁹ apud ARBAQUI et al., 2002). Podem ser entendidos como uma representação simplificada do processo de software.

A fim de fornecer mecanismos de apoio aos realizadores de processo de software, surgiram os Ambientes de Engenharia de Software Orientados a Processos (PSEE – *Process-Centered Software Engineering Environment*). Esses ambientes permitem a customização de um modelo de processo para um projeto específico por meio do refinamento e adaptação de um modelo de processo genérico (ARBAQUI et al., 2002). Alguns exemplos de PSEE são: *Eclipse Process Framework* - EPF (ECLIPSE FOUNDATION, 2008), *Rational Method Composer* (IBM, 2006), *Visual Studio Team System* - VSTS (MICROSOFT, 2006), TABA (MONTONI et al., 2004a, 2004b, 2005 e 2007), *Ontology-based Development Environment* - ODE (FALBO et al., 2004; SEGRINI et al., 2006) e *WebAPSEE* (LIMA et al., 2006).

Neste capítulo serão abordados alguns modelos de processo de software, um *framework* e uma norma que define os processos de ciclo de vida do software. Esses

⁸ CONRADI, R. C.; FERNSTROM, A.; FUGGETA, A.; SNOWDON, B. (1992). *Towards a Reference Framework for Process Concepts*. In: Proceedings of the 2nd European Workshop on the Software Process Technology, Lecture Notes in Computer Science, vol. 635, Trondheim, Norway.

⁹ LONCHAMP, J. (1993). *A Structured Conceptual and Terminological Framework for Software Process Engineering*. In: Proceeding of the 2nd International Conference on the Software Process, IEEE Computer Society Press, Berlin, Germany.

foram investigados para servirem de guia para uma MPE definir o seu processo padrão¹⁰.

2.2. Modelo Cascata

Trata-se do modelo mais antigo da engenharia de software (PRESSMAN, 2006; SOMMERVILLE, 2007) o qual sugere uma abordagem sistemática e seqüencial ao desenvolvimento de software (Figura 2.1). O enfoque desse modelo está nos documentos e nos artefatos.

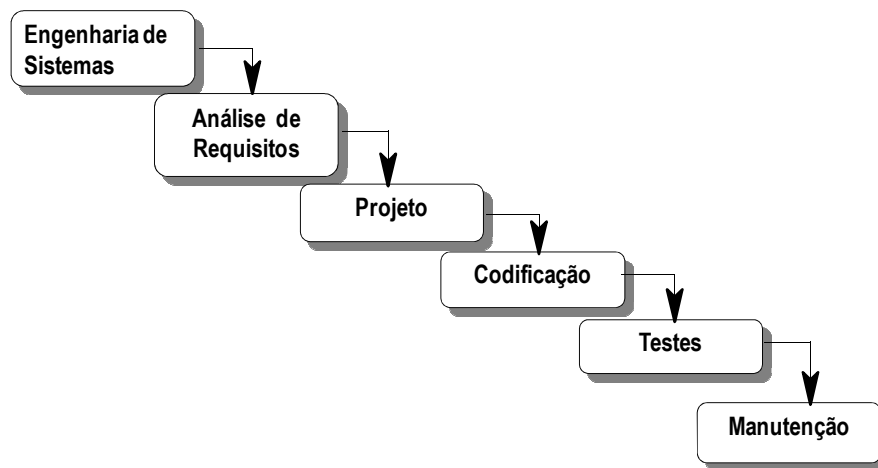


FIGURA 2.1 – MODELO DE PROCESSO DE SOFTWARE CLÁSSICO (CASCATA)

Porém, apesar de ser bastante conhecido, a utilização dele apresenta alguns problemas: projetos reais raramente seguem o fluxo seqüencial que o modelo propõe; logo no início é difícil estabelecer explicitamente todos os requisitos, pois no começo dos projetos sempre existe uma incerteza natural; o cliente deve ter paciência, pois uma versão executável do software só fica disponível em uma etapa avançada do desenvolvimento.

2.3. Modelo Incremental

O Modelo Incremental combina elementos do modelo cascata de maneira iterativa. Essa abordagem foi sugerida por Mills et al. (1987)¹¹ apud (PRESSMAN,

¹⁰ Segundo o CMMI, Processo Padrão refere-se à definição operacional do processo básico que guia o estabelecimento de um processo comum na organização.

2006) como um meio de reduzir o re-trabalho no processo de desenvolvimento de software e de proporcionar aos clientes algumas oportunidades de adiar decisões sobre seus requisitos detalhados, até que eles tenham alguma experiência com o sistema.

Os clientes identificam, em um esboço, as funções a serem fornecidas pelo sistema (Figura 2.2). Eles identificam quais funções são mais prioritárias e, em seguida, é definida uma série de estágios de entrega, com cada estágio fornecendo um subconjunto das funcionalidades do sistema, de acordo com a priorização do cliente.

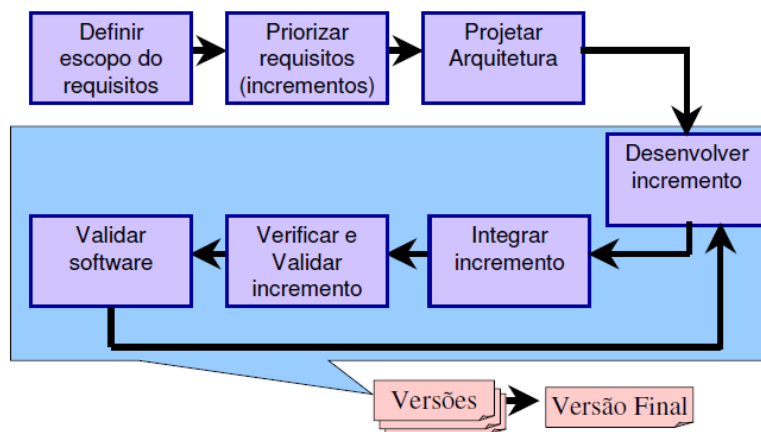


FIGURA 2.2 – MODELO INCREMENTAL

As vantagens desse modelo são:

- os clientes não precisam esperar que a versão final do software seja entregue para usá-lo;
- os clientes podem utilizar os primeiros incrementos desenvolvidos como um protótipo, de forma a definir melhor alguns requisitos do software;
- existe um risco menor de fracasso do software;
- como as funções prioritárias são entregues primeiro, é inevitável que estas passem por um período de testes mais intensivo.

¹¹ MILLS, H. D.; DYER, M.; LINGER, R. (1987). Cleanroom Software Engineering. IEEE Software, setembro de 1987, p. 19-25.

2.4. Modelo Espiral

O Modelo Espiral (Figura 2.3), originalmente proposto por Boehm (1988)¹² apud (BOEHM et al., 1998), usa uma abordagem cíclica para desenvolver um sistema, resultando em entregas incrementais da capacidade operacional do sistema.

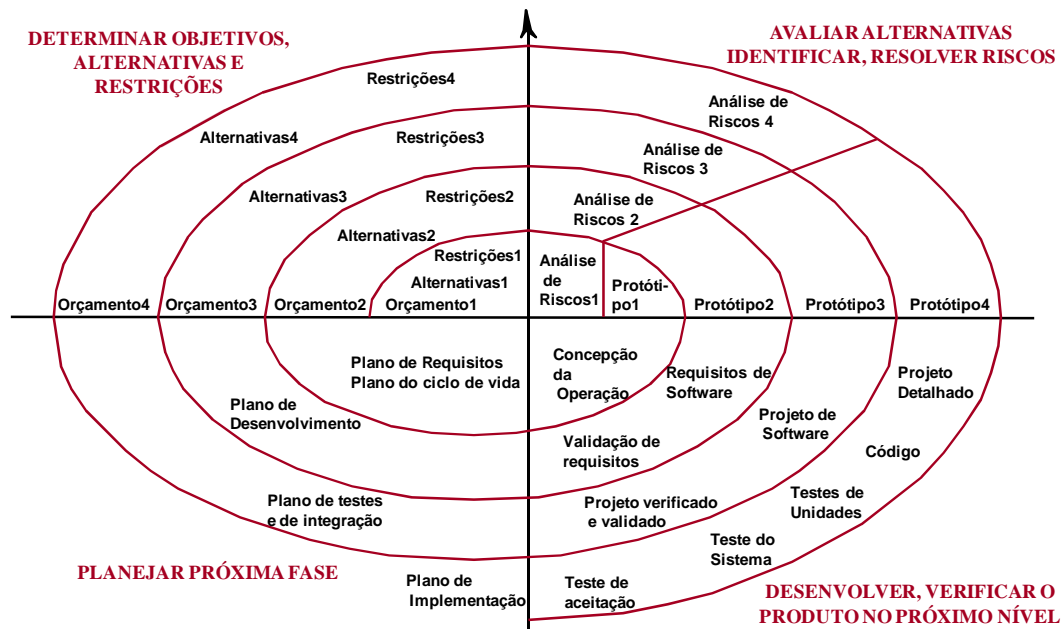


FIGURA 2.3 – MODELO ESPIRAL

Cada ciclo envolve quatro principais atividades:

- Elaborar os objetivos de processo e produto, restrições e alternativas.
- Avaliar as alternativas em relação aos objetivos e restrições; identificar e solucionar as principais fontes de risco do processo e produto.
- Elaborar a definição do produto e processo.
- Planejar o próximo ciclo e alterar o plano de ciclo de vida, incluindo a partição do sistema em subsistemas para ser distribuída em ciclos paralelos.

Pode-se dizer que o modelo espiral engloba os elementos do modelo cascata, com a característica de iteração do modelo incremental, adicionando um novo

¹² Boehm, B.W. (1988). A Spiral Model of Software Development and Enhancement. *Computer* 21, 5, 61–72.

elemento: a Análise de Risco. Pode ser criado um protótipo em qualquer etapa da evolução do produto, como mecanismo de redução de riscos (SOMMERVILLE, 2007).

2.5. RAD – Rapid Application Development

Trata-se de um modelo de processo de software incremental que enfatiza um ciclo de desenvolvimento curto (MARTIN, 1991¹³ apud SOMMERVILLE, 2007). Pode-se dizer que é uma adaptação do modelo cascata, onde o desenvolvimento rápido (em um período de 60 a 90 dias) é obtido devido à abordagem de construção ser realizada com base em componentes junto ao desenvolvimento realizado por várias equipes trabalhando em paralelo (Figura 2.4).

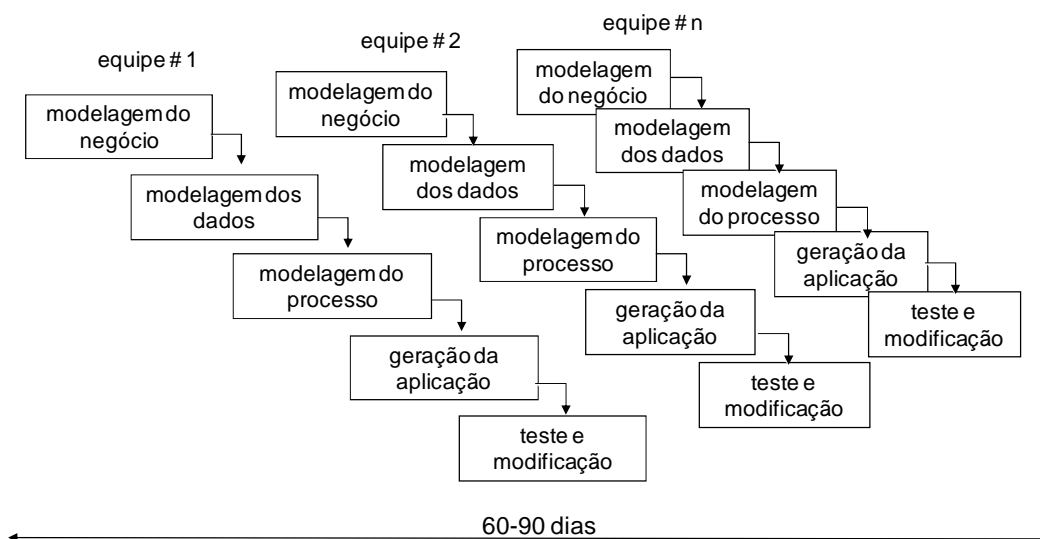


FIGURA 2.4 - MODELO RAD

Entretanto, para utilizar esse modelo é necessário que os requisitos sejam bem compreendidos, antes de iniciar a construção do software. Algumas desvantagens desse modelo: (1) para projetos grandes, o RAD exige recursos humanos suficientes para criar um número adequado de equipes RAD; (2) se os desenvolvedores e clientes não estiverem comprometidos com as atividades continuamente rápidas, o projeto

¹³ MARTIN, J. (1991). Rapid Application Development. Prentice-Hall.

falhará; (3) se o sistema não puder ser adequadamente modularizado, a construção dos componentes necessários será problemática; (4) o RAD pode não ser adequado quando riscos técnicos são altos (PRESSMAN, 2006).

2.6. Unified Process

O *Unified Process* (UP) é um *framework* genérico para o desenvolvimento de software que pode ser customizado para empresas ou projetos específicos. Esse *framework* é fundamentado em um conjunto de seis melhores práticas: desenvolver software de forma iterativa, gerenciar os requisitos, utilizar arquitetura baseada em componentes, modelar o software de forma visual, verificar a qualidade do software e controlar as mudanças do software (JACOBSON et al., 1999). O RUP (*Rational Unified Process*) foi originado do UP – elaborado pela *Rational Software Corporation* que foi adquirida pela IBM (KRUCHTEN, 2004).

Um dos princípios fundamentais do UP (Figura 2.4) é o fato de ser iterativo e incremental. Por isso, o projeto para desenvolvimento de sistema deve ser dividido em “mini-projetos”, chamados iterações.

Uma *iteração* é uma seqüência completa de disciplinas (*workflows*), sendo que o resultado de uma iteração é um artefato ou uma série de artefatos. Um artefato pode ser um código-fonte, um modelo, etc. Uma vez que os artefatos para uma fase específica foram completados ocorre a passagem para a próxima fase.

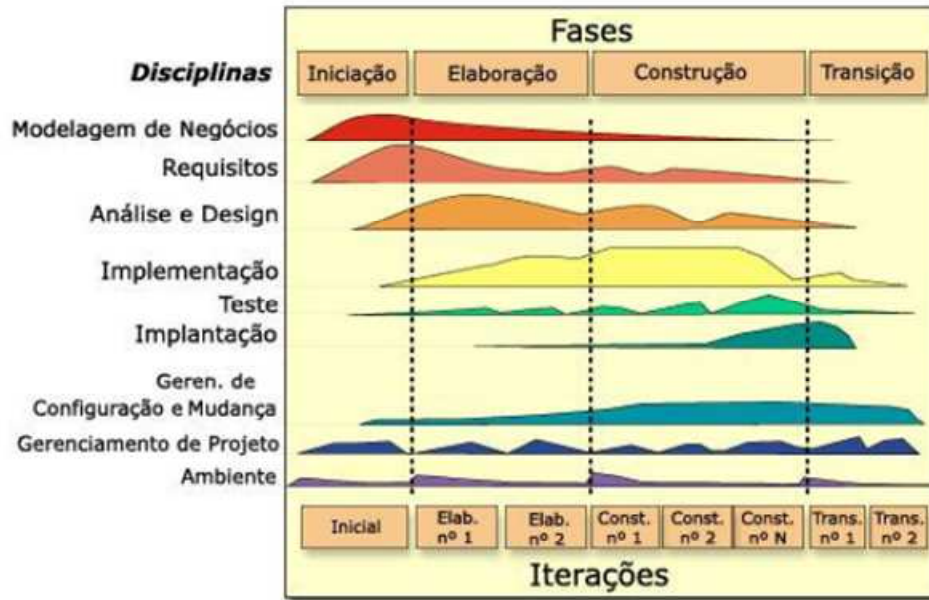


FIGURA 2.4 – UNIFIED PROCESS. FONTE: TRADUZIDO DE (AMBLER, 2005)

As fases do UP são (AMBLER, 2005):

- *Concepção*: nessa fase define-se o escopo do projeto, avalia-se a tecnologia, os principais riscos são relacionados, as áreas mais críticas a serem tratadas são detectadas, verifica-se a viabilidade do projeto e é desenvolvido um planejamento para a natureza incremental e iterativa do projeto.
- *Elaboração*: nessa fase os requisitos são especificados em detalhes e a arquitetura do software é identificada e avaliada.
- *Construção*: nessa fase o produto é construído e as suas versões beta são testadas.
- *Transição*: a versão do produto para entrega ao cliente é testada e o sistema é implantado.

2.7. Open/UP

OpenUP (ECLIPSE PROCESS FRAMEWORK COMMUNITY, 2008) é um Processo Unificado enxuto que aplica as abordagens iterativas e incrementais dentro de um ciclo de vida estruturado. Incorpora uma filosofia ágil e que foca a natureza colaborativa de desenvolvimento de software.

Em um projeto que utiliza OpenUP os esforços pessoais são organizados em micro-incrementos. Esses representam pequenas unidades de trabalho produzidas em poucas horas ou dias que fornecem *feedback* para direcionar decisões adaptativas em cada iteração (Figura 2.5).

OpenUP utiliza um ciclo de vida de iteração que estrutura como os micro-incrementos são aplicados para produzir entregas estáveis do sistema em semanas. Essas iterações são planejadas em um ciclo de vida de projeto composto por quatro fases: concepção, elaboração, construção e transição.

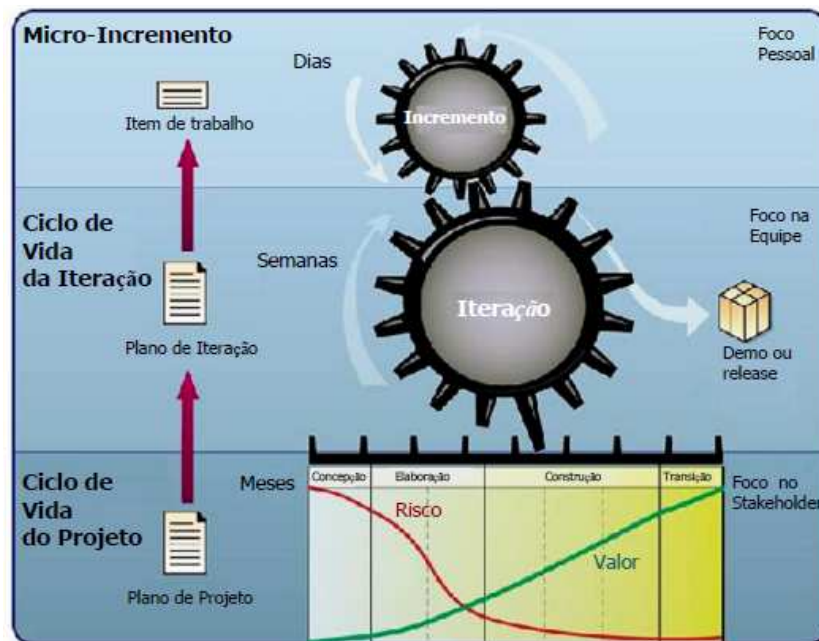


FIGURA 2.5 – OPENUP

FONTE: TRADUZIDO DE (ECLIPSE PROCESS FRAMEWORK COMMUNITY, 2008)

2.8. ISO/IEC 12207 – Processos de Ciclo de Vida de Software

Essa norma junto às suas emendas - ISO/IEC 12207 AMD1 e AMD2 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2001) - estabelece uma estrutura comum para os processos de ciclo de vida de software, que pode ser referenciada pela indústria de software. A estrutura contém processos, atividades e tarefas que servem para ser aplicadas durante (a) a aquisição de um sistema que contém software, de um produto de software independente ou de um serviço de software; (b) o fornecimento, desenvolvimento, operação e manutenção de produtos de software.

A norma não especifica o como implementar ou executar as atividades e tarefas, não determina um modelo de processo de software ou método de desenvolvimento e deve ser adaptada de acordo com o organização e projetos específicos.

Os processos de ciclo de vida estão organizados em categorias de processos fundamentais, processos de apoio e processos organizacionais (Figura 2.6). Cada categoria é subdividida em grupos de processo.

Os *Processos Fundamentais de Ciclo de Vida* constituem um conjunto de quatro grupos de processo que atendem as partes fundamentais (pessoa ou organização) durante o ciclo de vida de software: aquisição, fornecimento, engenharia e operação.

Os *Processos Organizacionais de Ciclo de Vida* constituem um conjunto de quatro grupos de processo que são empregados por uma organização para estabelecer e implementar uma estrutura subjacente, constituída de processos de ciclo de vida e pessoal associado, e melhorar continuamente a estrutura e os processos. Os grupos de processos organizacionais são: gerência, melhoria de Processo, recursos e infra-estrutura, e reuso.

Os *Processos de Apoio de Ciclo de Vida* constituem um conjunto de processos que auxiliam a um outro processo contribuindo para o sucesso e qualidade do projeto

de software. Um processo de apoio é empregado e executado, quando necessário, por outro processo. Os processos de apoio são: controle de configuração e garantia da qualidade.

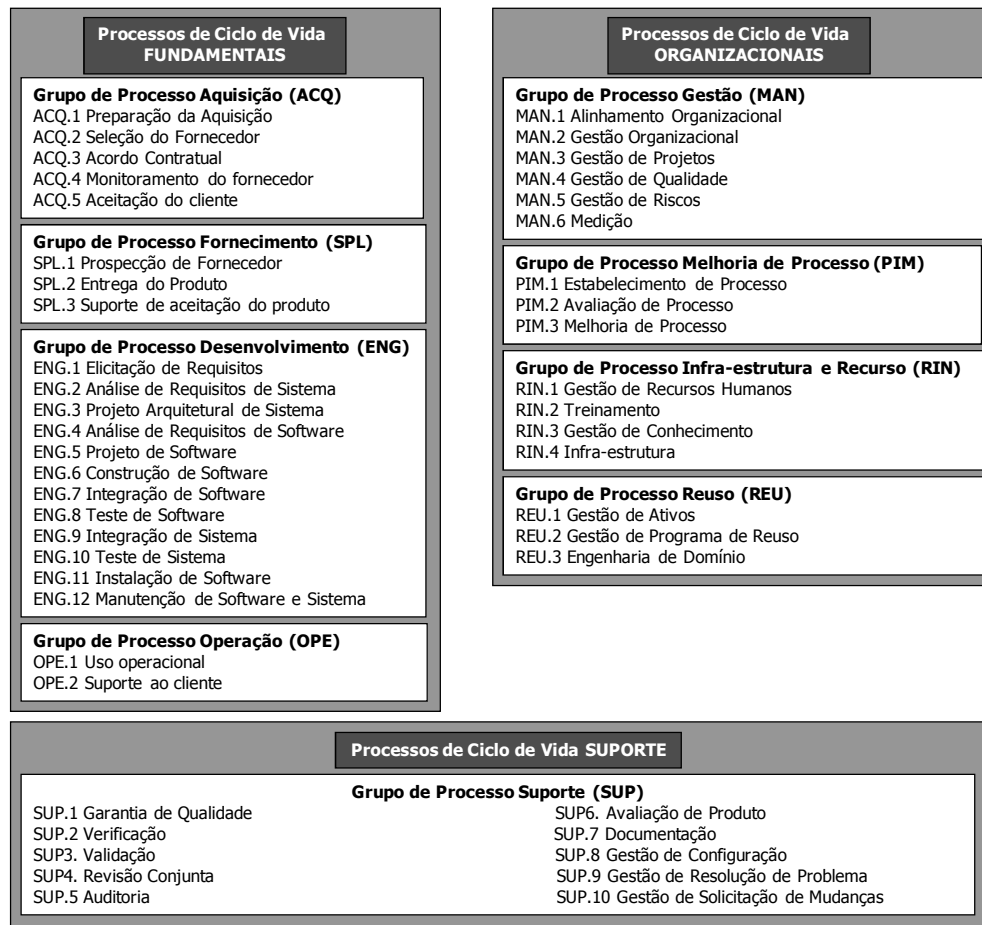


FIGURA 2.6 – ISO/IEC 12207 - PROCESSOS DE CICLO DE VIDA DE SOFTWARE
 FONTE: TRADUZIDO DE (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2001)

2.9. Considerações Finais

Os modelos cascata, espiral, incremental e RAD (PRESSMAN, 2006; SOMMERVILLE, 2007) são considerados modelos tradicionais e têm características de modelos prescritivos – modelos que prescrevem um conjunto de elementos de processo e tem ênfase em documentação e controles.

O Modelo Cascata não retrata fielmente o que acontece no desenvolvimento de projetos reais, devido sua característica de desenvolvimento seqüencial (conclui-se

uma fase para posteriormente iniciar a próxima). Essa deficiência foi suprida pelos modelos incremental e espiral que foram utilizados como base do *framework* de processo UP (*Unified Process*) (JACOBSON et al., 1999), pois sugere que o processo seja realizado de forma incremental e iterativa.

O modelo RAD é útil em situações peculiares: quando o desenvolvedor compreende todos os requisitos do software e re-utiliza componentes para agilizar o processo de desenvolvimento que deve ser realizado em um período de 60 a 90 dias.

Todos esses modelos (cascata, incremental, espiral e RAD), apesar de serem prescritivos, prescrevem apenas as atividades a serem realizadas sem fornecer mais detalhes (como os responsáveis pelas atividades, suas tarefas, os artefatos de entrada necessários para a realização das atividades, os artefatos resultantes da execução das atividades, recursos que podem ser utilizados e *templates* que fornecem informações do conteúdo de cada um dos artefatos), o que facilitaria o entendimento e uso do modelo por parte das MPEs.

O RUP - *Rational Unified Process* (KRUTCHEN, 2004) - embutido em algumas ferramentas da Rational e o *OpenUP* (ECLIPSE PROCESS FRAMEWORK COMMUNITY, 2008) – um processo unificado ágil embutido na ferramenta *free EPF Composer* (*Eclipse Process Framework*) – prescrevem elementos além das atividades a serem realizadas e poderiam servir como modelo a ser utilizado pelas MPEs, mas não incluem atividades relacionadas à comercialização de software, o que auxiliaria as MPEs durante a negociação da venda do software para o cliente e acompanhamento do fluxo de caixa e, também, não incluem atividades relacionadas ao gerenciamento da memória organizacional da empresa - Gestão de Conhecimento e nem atividades relacionadas à gestão de habilidades e competências dentro da empresa.

A norma ISO/IEC 12207 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2001) prescreve as atividades do processo de ciclo de vida de software, porém não define a seqüência de execução das atividades (por exemplo, linear ou evolutiva), o que é definido pelos modelos cascata, espiral, incremental,

framework UP e outros. Entretanto, pode ser utilizada como referência para a definição do processo de software padrão de uma empresa.

3. MELHORIA DE PROCESSO DE SOFTWARE

3.1. Considerações Iniciais

Melhorar continuamente os processos de negócio é um dos objetivos da Gestão de Processos de Negócio (BPM) - uma fonte de vantagem competitiva para a empresa (HUNG, 2006). Segundo Jeston & Nelis (2008) BPM tem por princípio alcançar os objetivos estratégicos da organização por meio da *melhoria*, *gestão* e *controle* de processos de negócio essenciais. A *melhoria* tem o objetivo de tornar os processos de negócio mais eficientes e eficazes. A *gestão* refere-se à medição e gerenciamento do desempenho dos processos e das pessoas. E, o *controle* é obtido por meio da medição dos processos de negócio em um ciclo de melhoria, tal como o PDCA (DEMING, 1986).

A melhoria do processo de software é considerada parte da engenharia de processo de software (EL EMAM, 2004). Esta pode ser examinada em dois níveis: o primeiro refere-se às atividades técnicas e gerenciais dentro dos processos de ciclo de vida que são realizados durante a aquisição, desenvolvimento, manutenção e retirada do software; o segundo é um meta-nível, referente à definição, implementação, avaliação, medição, gerenciamento, mudanças e melhorias dos próprios processos de ciclo de vida.

Segundo Salviano (2006), *“Melhoria de Processo de Software é uma abordagem para a melhoria de organizações intensivas em software, baseada em modelos de capacidade de processo de software, por meio do estabelecimento, avaliação e melhoria da capacidade de seus processos mais importantes, relacionados às atividades de aquisição, fornecimento, operação, desenvolvimento, manutenção, gerência, melhoria e/ou apoio de sistemas de software, com o objetivo de satisfazer de forma mais eficiente e eficaz os objetivos estratégicos da organização”*.

Há na literatura várias abordagens para melhoria de processo, que podem ser utilizadas como guia, tais como: PDCA (DEMING, 1986), IDEAL (MCFEELEY, 1996), Ciclo de Melhoria da 15504 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2003), Metodologia de Gestão de Mudanças (COSTA, 2006). Há, também, algumas abordagens elaboradas especificamente para o caso de MPEs, como PRO2PI-CYCLE (SALVIANO, 2006) e ASPE/MS (WEBER et al., 2005b), por terem orientação detalhada das atividades a serem realizadas (como fazer). Essas abordagens são comentadas neste capítulo.

A melhoria de processo utiliza um ou mais modelos como referência para orientar as ações de melhoria. Dentre esses modelos de referência¹⁴ estão os modelos de capacidade de processo, tais como OPM3 (PROJECT MANAGEMENT INSTITUTE, 2003), os modelos de capacidade do CMMI (SOFTWARE ENGINEERING INSTITUTE, 2006) e da ISO/IEC 15504 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2003), MR-MPS (WEBER et al., 2005a; SOFTEX, 2006) e outros modelos como PMBOK (PROJECT MANAGEMENT INSTITUTE, 2004), SWEBOOK (ABRAN et al., 2004) e ISO 9000 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2000). Alguns desses modelos de referência são comentados neste capítulo.

3.2. Gestão de Processos de Negócio – Conceitos Básicos

Processos de negócio são fatores chaves na integração de uma empresa (AGUILAR-SAVÉN & OLHAGER, 2002). Segundo Rozenfeld et al. (2006), “processos de negócio compreendem um conjunto de atividades organizadas entre si visando produzir um bem ou serviço para um tipo específico de cliente (interno ou externo à

¹⁴ Modelo de referência pode ser definido como um *framework* para entendimento de relacionamentos significativos entre as entidades de algum ambiente, e para o desenvolvimento de padrões ou especificações consistentes que apóiam o ambiente (Consultative Committee for Space Data Systems, Reference Model for an Open Archival Information System (OAIS), CCSDS 650.0-R-1.2, Red Book, June 2001) apud (SALVIANO, 2006).

empresa)”. Os processos de negócio podem representar operações repetitivas da empresa, que normalmente são estruturadas (exemplo: gestão financeira e gestão de produção). Outros processos de negócio não são tão estruturados, como é o caso do PDS (processo de desenvolvimento de software *on-demand*), uma vez que cada desenvolvimento de um software específico pode ser diferente.

Esses processos de negócio precisam ser gerenciados, de acordo com os objetivos estratégicos da organização para alcançar de modo eficiente esses objetivos. Por isso, é importante que estes processos estejam associados a uma abordagem de gestão – a Gestão de Processos de Negócio (BPM – *Business Process Management*). Essa abordagem pode ser visualizada tanto pela perspectiva de negócios quanto de tecnologia (SMITH & FINGAR, 2003). Considerando a perspectiva de negócios, BPM é considerada uma das melhores práticas dentre os princípios de gestão para auxiliar as organizações a sustentarem sua vantagem competitiva (KILMANN, 1995).

As raízes da BPM podem ser encontradas voltando na década de 80 com a filosofia de TQM (*Total Quality Management* – Gestão da Qualidade Total) e na década de 90 com a Reengenharia de Processo de Negócio (*Business Process Reengineering* - BPR). Os estudos de Davenport (1993) e Zairi & Sinclair (1995) indicam que TQM é de natureza incremental, evolucionária e contínua. Em oposição, BPR é de natureza radical, revolucionária e ocorre de uma vez só. A partir de uma perspectiva holística, BPM integra as abordagens TQM e BPR as quais podem ser consideradas como apropriadas para a melhoria de desempenho de processos de negócio na maioria das circunstâncias (HUNG, 2006).

BPM tem o objetivo de entender e organizar a empresa como processos de negócio (SMITH & FINGAR, 2003). Por outro lado, existem autores que enfatizam BPM como uma ferramenta tecnológica, vista como uma evolução de sistemas de gestão de *workflow* (AALST et al., 2003). Neste caso, o ciclo de vida de BPM tem quatro fases: projeto de processo, configuração de sistema, promulgação e diagnóstico. O foco de sistemas tradicionais de gestão de *workflow* é nas fases de

projeto de processo e promulgação de processo do ciclo de vida de BPM. Esta definição de BPM estende a tradicional abordagem de gestão de *workflow* para suportar a fase de diagnóstico e permite por meio de novos caminhos apoiar o processo operacional (AALST et al., 2003; AALST, 2004).

No presente trabalho de pesquisa o escopo é considerar a melhoria do modelo de Processo de Venda e Desenvolvimento de Software (PV&DS) *on-demand* em um contexto de BPM mais abrangente - o foco é sobre a visão de negócio de BPM. De qualquer modo, a implementação de tecnologia BPM é uma questão muito importante, tanto que é considerada em sistemas de Gestão de Ciclo de Vida de Produto (*Product Lifecycle Management* – PLM).

Para Zairi (1997) essa abordagem é dependente de elementos estratégicos, elementos operacionais, do uso de ferramentas e técnicas modernas, do envolvimento de pessoas e, o mais importante, tem um foco horizontal que possibilita o atendimento dos requisitos do cliente da melhor forma possível. Em BPM, o alinhamento das operações de negócio com as prioridades estratégicas e o envolvimento de pessoas são conceitos fundamentais na busca pela competitividade (ZAIRI, 1997; MCKAY & RADNOR, 1998).

Hung (2006) entende BPM como uma filosofia de gestão integrada e um conjunto de práticas que incluem mudanças incrementais e radicais no processo de negócio, e enfatiza a melhoria contínua, satisfação do cliente e envolvimento dos funcionários. Nesse contexto, a gestão de mudanças se integra à abordagem BPM.

Zairi (1997) afirma que o BPM tem que ser governado por alguns princípios, dentre eles: (1) as principais atividades têm que ser devidamente mapeadas e documentadas; (2) BPM cria um foco nos clientes por meio de ligações horizontais entre as atividades; (3) BPM é dependente de sistemas e procedimentos documentados para assegurar disciplina, consistência e repetibilidade no desempenho da qualidade; (4) BPM é dependente de atividades de medição para avaliar o

desempenho de cada processo individualmente; (5) BPM tem base em uma abordagem contínua de otimização e (6) BPM é uma abordagem de mudança cultural.

A fim de complementar a definição dos princípios da BPM, Hung (2001)¹⁵ apud Hung (2006) inclui: a visão holística, a autoridade estratégica, a TI (tecnologia da informação) que permite o gerenciamento dos processos de negócio, a transformação do negócio, o impacto por toda a organização e a ênfase inter-funcional da gestão do processo.

3.3. Algumas Abordagens para Melhoria de Processo

3.3.1. PDCA

A idéia original do PDCA (*Plan, Do, Check, Action*) foi desenvolvida na década de 1930 por Shewhart e popularizado por Deming (1986) como sendo um ciclo de controle estatístico do processo que pode ser repetido continuamente sobre qualquer processo ou problema. O início do ciclo PDCA se dá com o planejamento do que será feito e com a definição de metas e métodos para atingir essas metas. Este plano é implantado, coletando-se dados e é realizada a verificação dos resultados em relação às metas definidas. Posteriormente, são tomadas ações corretivas ou de melhoria, caso necessário.

3.3.2. IDEAL

O modelo IDEAL (*Initiating, Diagnosing, Establishing, Acting, Learning*) foi criado pelo SEI (*Software Engineering Institute*) para o aprimoramento contínuo do processo de desenvolvimento de software (MCFEELEY, 1996). Foi desenvolvido para ser uma ferramenta de apoio à implantação do CMM (*Capability Maturity Model*) para software.

¹⁵ HUNG, R. Y. (2001). **An empirical examination of the relationship between BPM and business performance**: a study of Australia's top 1000 companies (unpublished PhD dissertation, The University of Sydney).

Trata-se de um modelo de programa de melhoria de processo de software (SPI – *Software Process Improvement*) usado para guiar o desenvolvimento de um plano integrado para inicialização e gerenciamento de um programa de SPI. Esse modelo é composto por 5 fases que compõem o ciclo de melhoria. É importante ressaltar que o tempo para completar um ciclo por completo pode variar de organização para organização.

O ciclo de melhoria começa com a fase de *Inicialização* que foca o comprometimento dos recursos iniciais e o estabelecimento da infra-estrutura necessária para a melhoria. A segunda fase é o *Diagnóstico* que estabelece os níveis atuais de maturidade de processo, a descrição atual do processo e inicia o desenvolvimento de um plano de ação. A terceira fase é o *Estabelecimento* com o objetivo de estabelecer metas e prioridades e completar o plano de ação. A quarta fase é a *Ação* na qual realiza-se a pesquisa e são desenvolvidas soluções para os problemas do processo, expandindo as melhorias de processo com sucesso para toda a organização. Por fim, a fase *Alavancagem* tem o objetivo a preparação para o próximo ciclo, a partir da fase de Diagnóstico, e aplicação das lições aprendidas para refinar o processo de SPI.

3.3.3. Ciclo de Melhoria da ISO/IEC 15504

A parte 4 da norma ISO/IEC 15504 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2003) descreve orientações para utilização de modelos para melhoria de processo ou determinação da capacidade de processo.

O guia de melhoria de processo é composto por 8 (oito) etapas. Na primeira etapa examinam-se os objetivos de negócio da organização e a motivação existente para melhorar. Na segunda etapa o programa de melhoria do processo é iniciado; ele deve ser implementado e gerenciado como um projeto com definição de responsáveis, orçamento, cronograma, *milestones*, entre outros. Na terceira etapa avalia-se a

capacidade atual do processo, utilizando como referência as partes 2 (*framework* de avaliação) e 3 (guia de execução da avaliação) da ISO/EC 15504. Na quarta etapa, o resultado da avaliação é analisado e confrontado com os objetivos de negócios da organização para: identificar, analisar e listar as áreas de melhoria; definir os objetivos específicos da melhoria e derivar um plano de ação. Na quinta etapa, o plano de ação é implementado para melhorar os processos da organização. Essa implementação pode ser simples ou complexa, dependendo do conteúdo do plano de ação e das características da organização. Em geral, vários projetos de implementação podem ser iniciados, cada um concentrado em implementar uma ou mais ações definidas no plano de ação. Na sexta etapa, quando os projetos de implementação foram realizados, a organização deve: confirmar se os objetivos planejados foram alcançados e se os benefícios esperados se concretizaram; checar se os processos e práticas apropriados foram adotados; confirmar que a cultura organizacional foi modificada, conforme apropriado; re-avaliar os riscos associados ao programa de melhoria do projeto e re-avaliar os custos e benefícios. Na sétima etapa devem ser sustentadas as melhorias, mantendo a melhoria do nível de capacidade alcançado. Isso requer gerenciamento para monitorar a institucionalização dos processos melhorados. A última etapa tem como objetivo monitorar continuamente o desempenho dos processos da organização e iniciar melhorias em novos processos, como parte do programa de melhoria contínua.

3.3.4. Metodologia de Gestão de Mudanças

A Metodologia de Gestão de Mudanças foi concebida a partir da sistematização das atividades de métodos de gestão de mudança analisados por Costa (2006).

Na primeira fase da metodologia (Figura 3.1), “Definição da Mudança”, avalia-se a necessidade de mudança e seus objetivos juntamente à estratégia empresa. Na fase de “Diagnóstico” é realizada a análise da situação atual. Na fase seguinte, “Definir

portfólio de projetos”, as propostas de projetos de mudanças existentes são analisadas e suas implantações são priorizadas. É importante ressaltar que essa fase só ocorre quando existem vários projetos de mudança.

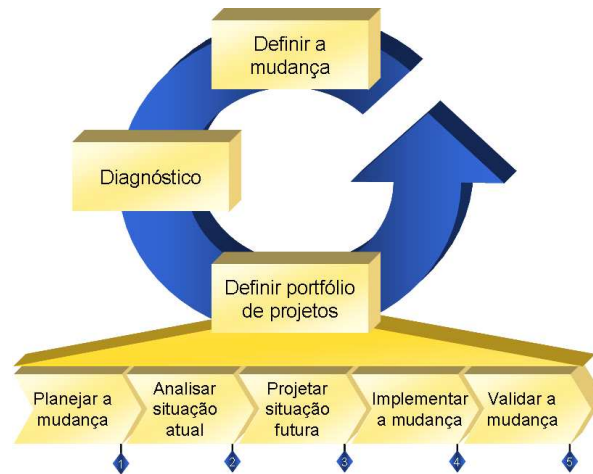


FIGURA 3.1 – METODOLOGIA DE GESTÃO DE MUDANÇAS. FONTE: (COSTA, 2006)

Para cada um dos projetos selecionados, é dado início à fase “Planejar a mudança”, em que por meio do escopo do projeto, são definidos e refinados os objetivos da mudança e planejadas as ações necessárias para se alcançar esses objetivos. Nessa fase, também são elaborados a WBS (*work breakdown structure*) e o cronograma do projeto. Na fase “Analisar a situação atual”, é feito um diagnóstico mais detalhado da situação corrente, levantados os requisitos da mudança e os dados quantitativos e qualitativos da situação em que o projeto de melhoria está intervindo.

Na fase seguinte, “Projetar a situação futura”, é projetada a solução, considerando-se os objetivos iniciais do projeto de melhoria e a opinião dos interessados (*stakeholders*) no projeto de mudança, para que as possíveis resistências sejam minimizadas. Na fase “Implementar a mudança” são realizados os treinamentos e é implantada a mudança e comunicada a todos os envolvidos. Por fim, na fase “Validar a mudança”, após um período de institucionalização da mudança, é verificado se houve algum desvio na implantação. É analisado se o problema inicial e a

necessidade de mudança foram satisfeitos e, por fim, comunicado o encerramento do projeto em questão.

3.3.5. PRO2PI-CYCLE

Salviano (2006), em sua pesquisa, caracteriza as diferenças e similaridades entre as arquiteturas estagiada e contínua para modelos de capacidade de processo e faz reflexões sobre tais arquiteturas para subsidiar uma evolução da atual melhoria de processo para uma melhoria mais viável e mais adequada ao contexto e objetivos estratégicos da organização: uma Engenharia de Processo Dirigida por Perfil de Capacidade de Processo (*Process Capability Profile Driven Process Engineering – PDCE*). A partir dessas reflexões, identifica três gerações de arquiteturas de modelos de capacidade de processo. São elas:

- *Arquitetura estagiada fixa (1ª geração)*: os modelos desta geração são compostos por uma seqüência de PCP (perfil de capacidade de processo), onde cada um é composto por uma combinação fixa de áreas de processo em um determinado nível de capacidade de processo. A utilização de um modelo deste tipo deve seguir exatamente o que ele propõe, podendo apenas ser interpretadas as orientações para o contexto da utilização. Exemplos: modelo na representação estagiada do CMMI, MR-MPS (Modelo de Referência) do MPS.BR.

- *Arquitetura contínua fechada (2ª geração)*: as áreas de processo e os níveis de capacidade de processo são fixos e as suas combinações são variáveis. A utilização de um modelo com este tipo de arquitetura passa pela escolha de um PCP, a partir dos níveis de capacidade de processo e áreas de processo, seguindo as regras de composição. Exemplos: ISO/IEC 15504-5 e modelo na representação contínua do CMMI.

- *Arquitetura contínua aberta (3ª geração)*: apenas os níveis de capacidade são fixos. Com isto, podem ser definidos conjuntos de áreas de processo e, então,

definidas combinações de áreas de processo e níveis de capacidade de processo em PCPs. A utilização de modelos com este tipo de arquitetura passa por uma etapa anterior às da utilização de modelos com arquitetura da 2ª geração, pois inclui a escolha ou definição de áreas de processo. Assim, aumenta-se a flexibilidade e as oportunidades de melhor alinhamento da melhoria com a organização. Entretanto, aumenta-se o risco de escolhas inadequadas, o que leva ao aumento da necessidade de metodologias para sua utilização.

A evolução da atual melhoria de processo - PDCE, proposta por Salviano (2006), é fundamentada na utilização da 3ª geração de arquitetura de modelos de capacidade – contínua aberta. Como representante dessa geração, Salviano (2006) propõe a extensão do ISO/IEC 15504. Ele defende que, em geral, a melhoria de processo pode ser baseada em PCPs dinâmicos e específicos, ou ainda, em níveis de maturidade dinâmicos e específicos. Assim, é diluída a separação rigorosa entre definição de modelos (hoje feita mais por institutos) e a utilização de modelos, uma vez que uma organização ao fazer a melhoria de processo, também define modelos por meio de PCPs.

O “estabelecimento de perfil de capacidade de processo” pode ser visto como uma área de processo, segundo o CMMI, ou como um processo do grupo de melhoria de processo – PIM (*Process Improvement Process Group*) da norma ISO/IEC 12207 e do modelo ISO/IEC 15504-5.

Para experimentar, validar e evoluir a Engenharia de Processo Dirigida por Perfil de Capacidade de Processo (PDCE), Salviano (2006) propõe a abordagem PRO2PI¹⁶ (Perfil de Capacidade de Processo para Melhoria de Processo).

PRO2PI-CYCLE é um processo para ciclo de melhoria com PRO2PI, composto por seis fases: 1. Inicia o ciclo de melhoria, 2. Avalia práticas correntes, 3. Planeja

¹⁶ PRO2PI: é uma abstração, segundo o aspecto de capacidade de processo, de um processo, e pode ser utilizado como referência para a melhoria do processo atual para o estado estabelecido por ele, em uma organização.

ações de melhoria, 4. Realiza ações de melhoria, 5. Prepara a institucionalização da melhoria e 6. Institucionaliza a melhoria.

Para tornar viável para MPEs intensivas em software, com processos de baixa capacidade (que buscam até o nível 3 de capacidade, segundo a 15504, por exemplo), iniciarem um ciclo de melhoria, estabelecendo um PRO2PI, Salviano (2006) definiu um método, denominado PRO2PI-WORK (*PRO2PI Establishment Workshop Method*). Esse método realiza de forma compactada as atividades das três primeiras etapas do ciclo de melhoria PRO2PI-CYCLE.

3.3.6. ASPE/MSC

ASPE/MSC (*Approach for Software Process Establishment in Micro and Small Companies*) tem o objetivo de estabelecer processos de software voltada para MPEs, de forma incremental, visando a melhoria contínua (WEBER et al., 2005b).

Para o estabelecimento de processos de software no contexto de MPEs é necessário que haja baixo investimento, fácil compreensão, orientação detalhada das atividades a serem executadas (como fazer), ser de domínio público, suporte de guias para sua implantação e adaptação, entre outros.

As fases dessa abordagem são: 1. Diagnóstico do Processo de Software Atual, obtido por meio de um método de avaliação específico para MPEs); 2. Análise Estratégica para definição e priorização das ações para o estabelecimento de processos na empresa; 3. Definição do Processo, de forma explícita e descritiva para que as pessoas se orientem na execução das atividades; 4. Implantação do Processo por meio da institucionalização e avaliação do processo que foi definido na organização.

3.4. Modelos de Referência utilizados para Melhoria de Processo

Alguns modelos de referência existentes na literatura, tanto modelos de capacidade de processo quanto outros modelos que podem ser usados como referências para a melhoria de processo são apresentados, a seguir.

3.4.1. Framework ISO/IEC 15504 e Modelo 15504-5

A ISO/IEC 15504 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2003) é um *framework* de avaliação de processo genérico para qualquer área de tecnologia, inclusive a engenharia de software, com dois contextos de uso: Melhoria do Processo e Determinação da Capacidade de Processo. No primeiro contexto, o resultado identifica a capacidade do processo que é utilizada para identificar mudanças de melhoria. Já no segundo contexto, a capacidade identificada é utilizada para descobrir riscos que o processo avaliado incorre e, também, pode ser utilizado como motivação para a melhoria do processo.

A norma ISO/IEC 15504 está descrita em cinco partes: a parte 1 refere-se aos conceitos, a parte 2 descreve o *framework* de avaliação, a parte 3 descreve orientações para uso do *framework* descrito na parte 2 e a parte 4 descreve orientações para utilização de modelos para melhoria ou determinação da capacidade. A parte 5 - 15504-5 (*Information Technology — An exemplar Process Assessment Model*) - refere-se a um modelo de avaliação exemplo que apóia a realização de uma avaliação por meio de indicadores que orientam na interpretação dos objetivos e resultados definidos nas emendas Amd 1 e Amd 2 da ISO/IEC 12207 e dos atributos de processo definidos na 15504-2. Esse modelo aborda a área de engenharia de software, embora a ISO/IEC 15504 possa ser utilizada em qualquer área tecnológica.

A arquitetura definida pela ISO/IEC 15504 é contínua, com duas dimensões: uma que define os processos (dimensão de processo) e outra que define os níveis de

capacidade que cada um dos processos pode ter ao serem estabelecidos na organização (dimensão de capacidade).

A dimensão de processo usa as definições de processos das emendas Amd1 e Amd2 da ISO/IEC 12207 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2001) para identificar um Modelo de Referência de Processo. Os processos do Modelo de Referência são classificados em três categorias: fundamentais do ciclo de vida, de apoio ao ciclo de vida e organizacionais. Os *processos fundamentais* são essenciais durante o ciclo de vida do software e estão agrupados em: processos de aquisição, de fornecimento, de desenvolvimento e de operação. Os *processos organizacionais* são utilizados para estabelecer e implementar uma estrutura subjacente na organização, e também, para melhorar continuamente a estrutura e os processos. Esses processos estão agrupados em: processos de gerenciamento, de melhoria de processo, de infraestrutura e recursos, e de reuso. Já os *processos de apoio ao ciclo de vida* consistem de processos que dão suporte a outros processos como uma parte integrante, com um propósito distinto, e contribui para o sucesso e qualidade do projeto de software.

Cada processo no Modelo de Avaliação é descrito em termos da declaração de um *propósito*. A lista de *resultados* é associada a cada declaração do propósito do processo, como uma lista de resultados positivos esperados da realização, ou melhor, do desempenho do processo. A satisfação do propósito de um processo representa o primeiro passo em direção ao nível 1 de capacidade de processo onde os resultados esperados são observáveis.

O Modelo de Avaliação de Processo expande as definições de processo do Modelo de Referência de Processo incluindo um conjunto de indicadores de desempenho de processo: *práticas base* e *produtos de trabalho*. Uma *prática base* é uma atividade direcionada ao propósito de um processo que identifica “o quê” deve ser feito, sem especificar o “como”. Para cada processo, de acordo com a *dimensão processo*, há um conjunto de práticas base definidas ao qual, uma vez realizado,

proporciona o alcance dos resultados que refletem o propósito do processo; cada *prática base* é explicitamente associada a um *resultado* do processo. A realização de um processo produz *produtos de trabalho* que são identificáveis e utilizados para alcançar o propósito do processo; cada *produto de trabalho* de entrada e saída é relacionado a um ou mais *resultados*. As *práticas base* e *produtos de trabalho* são usados para medir o grau de alcance do atributo *desempenho do processo* avaliado, ou seja, dão suporte à avaliação de um processo em relação ao nível 1 de capacidade. Cada *produto de trabalho* tem um conjunto de *características do produto* que o identifica e pode ser usado como referência na revisão do produto de trabalho a fim de avaliar a realização/desempenho efetivo de um processo.

A *dimensão de capacidade* do Modelo de Avaliação de Processo utiliza as definições de *níveis de capacidade* e *atributos de processo* da 15504-2 e expande cada um dos nove atributos por meio da inclusão de um conjunto de indicadores de capacidade de processo (*práticas genéricas*, *recursos genéricos* e *produtos de trabalho genéricos*). *Atributos de processo* (PA – *process attributes*) são características de um processo que podem ser avaliadas de acordo com uma escala de desempenho, fornecendo uma medida da capacidade do processo. Cada atributo mede um aspecto particular da capacidade do processo necessário para apoiar a melhoria do processo e a determinação da capacidade. Um *nível de capacidade* é um conjunto de atributos de processo que trabalham juntos para fornecer uma alavancagem na capacidade de desempenho de um processo. Esses níveis constituem um caminho racional de progresso por meio da melhoria da capacidade de algum processo.

As *práticas genéricas* (GP) são atividades genéricas que fornecem um guia para a implementação dos *atributos de processo*; muitas delas são práticas gerenciais. Os *recursos genéricos* (GR) são pessoas, ferramentas, métodos e infraestrutura que podem ser usados na realização dos *atributos de processo*. E, os *produtos de trabalho genéricos* (GWP) são conjuntos de características evidenciadas nos produtos de

trabalho como resultado da realização de um atributo. Os GWP são identificados de 01-00 à 21-00; para cada um deles há produtos de trabalho específicos (SWP) identificados pelos dois primeiros dígitos do identificador do GWP seguido de dois dígitos seqüenciais. Exemplo: 05-00 é o identificador do GWP “objetivos do negócio” e 05-01, 05-02, 05-03 são exemplos de identificadores de SWP desse GWP.

Há seis níveis de capacidade, incorporando nove atributos de processo. E, para cada um desses atributos existem práticas genéricas (GP), recursos genéricos e produtos de trabalho genéricos. Os níveis de capacidade são:

- Nível 0: Processo incompleto: o processo não é implementado, ou falha na tentativa de alcançar os propósitos do processo. Neste nível, há poucas ou nenhuma evidência de alguma realização sistemática do propósito do processo.
- Nível 1: Processo executado: o processo implementado atinge ao propósito do processo.
- Nível 2: Processo gerenciado: o processo realizado (nível 1) é agora implementado em um estilo gerenciado (planejado, monitorado e ajustado) e os *produtos de trabalho* são apropriadamente estabelecidos, controlados e mantidos.
- Nível 3: Processo estabelecido: o processo gerenciado (nível 2) é agora implementado usando um processo definido que é capaz de alcançar os resultados do processo.
- Nível 4: Processo previsível: o processo estabelecido (nível 3) opera dentro de limites definidos para alcançar os resultados do processo.
- Nível 5: Processo em otimização: o processo previsível (nível 4) é melhorado continuamente para atingir objetivos de negócio projetados e relevantes.

3.4.2. Framework SEI CMMI e Modelo CMMI-DEV

O propósito do *framework* CMMI (*Capability Maturity Model Integration*) é fornecer orientação para melhorar o processo da organização e gerenciar o desenvolvimento, aquisição e manutenção de produtos e serviços (SOFTWARE ENGINEERING INSTITUTE, 2006).

A arquitetura do modelo CMMI foi melhorada para apoiar várias constelações e o compartilhamento das melhores práticas entre as constelações e os seus modelos.

Uma constelação CMMI é um conjunto de componentes que inclui um modelo, o seu material didático, e métodos de avaliação para apoiar uma área de interesse. Atualmente, existem três constelações: desenvolvimento (CMMI-DEV), serviços (CMMI-SVC) e aquisição (CMMI-ACQ). Os modelos do CMMI que já estavam disponíveis na comunidade antes de 2006 foram considerados parte da constelação do CMMI para o Desenvolvimento (CMMI-DEV).

CMMI-DEV é um modelo de referência que consiste de melhores práticas que abordam atividades de desenvolvimento e manutenção aplicadas a produtos e serviços. Ele contém práticas que cobrem gestão de projetos, gestão de processos, engenharia de sistemas, engenharia de hardware e outros processos de apoio usados no desenvolvimento e manutenção, substituindo os modelos CMMI-SE/SW da versão anterior.

A constelação CMMI-DEV consiste em dois modelos: CMMI for *Development* e CMMI for *Development* +IPPD (contém objetivos e práticas adicionais que cobrem o uso de equipes integradas nas atividades de desenvolvimento e manutenção).

Os componentes do modelo são agrupados em três categorias: requeridos, esperados e informativos (Figura 3.1).

Os componentes requeridos descrevem o quê uma organização deve alcançar para satisfazer à área de processo¹⁷ (PA – *process area*). Os componentes esperados descrevem o quê uma organização pode implementar para alcançar um componente requerido. Os componentes informativos fornecem detalhes que auxiliam as organizações a pensarem em como abordar os componentes requeridos e esperados. As sub-práticas e os produtos de trabalho são exemplos de componentes informativos.

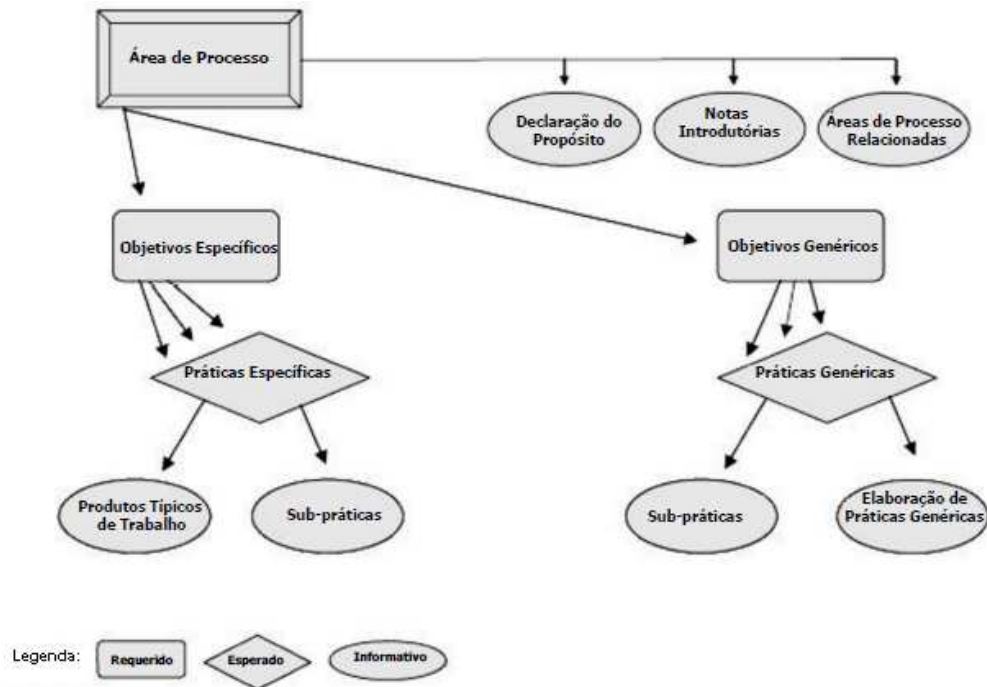


FIGURA 3.1 – COMPONENTES DOS MODELOS DO CMMI
FONTE: TRADUZIDO DE (SOFTWARE ENGINEERING INSTITUTE, 2006)

A arquitetura dos modelos do CMMI é composta basicamente por áreas de processo, organizadas em duas representações: contínua e por estágios.

A arquitetura por estágios utiliza-se dos níveis de maturidade para representar a melhoria em um conjunto de áreas de processo e, por conseqüência, na organização. Já a arquitetura contínua utiliza-se dos níveis de capacidade para representar a melhoria em áreas de processo selecionadas. Por isso, os níveis de

¹⁷ Uma área de processo é um grupo de práticas relacionadas em uma área que, quando implementadas coletivamente, satisfazem um conjunto de objetivos considerados importantes para realizar a melhoria nessa área.

maturidade de 2 a 5 usam os mesmos termos que os níveis de capacidade de 2 a 5. Isso é intencional devido os conceitos de níveis de maturidade e capacidade serem complementares. Os níveis de maturidade, de 1 a 5, são:

1. *Inicial*: no nível de maturidade 1, os processos são usualmente *ad hoc* e caóticos. A organização usualmente não fornece um ambiente estável para apoiar os processos. O sucesso nessas organizações depende da competência de pessoas da organização e não do uso de processos comprovados. Apesar desse caos, as organizações desse nível de maturidade freqüentemente produzem produtos e serviços que funcionam; de qualquer modo, elas freqüentemente excedem seus orçamentos e não atendem aos seus cronogramas. Além disso, são caracterizadas por abandonar seus processos em tempo de crise e pela não habilidade de repetir seus sucessos.

2. *Gerenciado*: no nível de maturidade 2, os projetos da organização tem assegurado que os processos são planejados e executados de acordo com uma política; os projetos empregam pessoas habilitadas que tem recursos adequados para produzir os resultados controlados; envolvem stakeholders relevantes; são monitorados, controlados e revisados; e são avaliados pela aderência às descrições de seus processos. Os estados dos produtos de trabalho e da entrega de serviços são visíveis para o gerenciamento nos pontos definidos (ex, em *milestones*). Os compromissos são estabelecidos entre os stakeholders relevantes e são revisados quando necessário.

Os produtos de trabalho são apropriadamente controlados e tanto os produtos de trabalho quanto os serviços satisfazem suas descrições de processo, padrões e procedimentos.

3. *Definido*: no nível de maturidade 3, os processos são bem caracterizados e compreendidos, e são descritos em padrões, procedimentos, ferramentas e métodos. O conjunto de processos padrão da organização, os quais são base para a maturidade no nível 3, é estabelecido e melhorado com o passar do

tempo. Esses processos padrão são usados para estabelecer consistência através da organização. Os projetos estabelecem seus processos definidos a partir desse conjunto de processos padrão. Nesse nível, o objetivo genérico 3 e suas práticas genéricas (não endereçadas no nível de maturidade 2) são aplicados nas PAs do nível de maturidade 2 e 3, para alcançar o nível de maturidade 3.

4. Gerenciado Quantitativamente: no nível de maturidade 4, a organização e os projetos estabelecem os objetivos quantitativos para a qualidade e desempenho dos processos e usa-os como critérios para o gerenciamento dos processos. Os objetivos quantitativos são baseados nas necessidades do cliente, usuários finais, organização e implementadores de processo. A qualidade e o desempenho do processo são entendidos em termos estatísticos e é gerenciado ao longo da vida do processo.

Para sub-processos selecionados, as medidas detalhadas do desempenho do processo são coletadas e analisadas estatisticamente. Essas medidas são incorporadas em um repositório de medidas da organização para apoiar na tomada de decisão.

As causas especiais de variação de processo são identificadas e, quando apropriadas, as origens das causas especiais são corrigidas para a prevenção de ocorrências futuras. A diferença entre os níveis de maturidade 3 e 4 é a previsibilidade do desempenho do processo; no nível de maturidade 4, o desempenho dos processos é controlado usando técnicas estatísticas e outras técnicas quantitativas, e é previsível quantitativamente. No nível de maturidade 3, os processos são tipicamente previsíveis somente qualitativamente.

5. Em otimização: no nível de maturidade 5, uma organização melhora continuamente seus processos com base no entendimento quantitativo das causas comuns de variação inerentes no processo. Esse nível de maturidade

foca na melhoria contínua do desempenho de processo por meio de melhorias tecnológicas e de processo incrementais e inovadoras. Os objetivos de melhoria de processo quantitativo para uma organização são estabelecidos, continuamente revisados para refletir as mudanças dos objetivos do negócio, e usados como critérios para o gerenciamento da melhoria do processo. Os efeitos das melhorias de processo implantadas são medidos e avaliados em relação aos objetivos de melhoria quantitativa de processo. Tanto os processos definidos quanto o conjunto de processos padrão da organização são alvos de atividades de melhoria mensuráveis.

A diferença entre os níveis de maturidade 4 e 5 é o tipo de variação de processo endereçada. No nível de maturidade 4, a organização está preocupada em endereçar causas especiais de variação de processo e fornecer previsibilidade estatística dos resultados. Embora os processos possam produzir resultados previsíveis, os resultados podem ser insuficientes para alcançar os objetivos estabelecidos. No nível de maturidade 5, a organização está preocupada em endereçar causas comuns de variação do processo e mudar o processo a fim de melhorar o desempenho do processo e alcançar os objetivos quantitativos de melhoria do processo.

Os níveis de capacidade de 2 a 5 usam os mesmos termos que os objetivos genéricos de 2 a 5. Isso é intencional devido a cada um dos objetivos e práticas genéricas refletirem o significado dos níveis de capacidade em termos de objetivos e práticas que podem ser implementadas. Os seis níveis de capacidade, de 0 a 5, são:

0. Incompleto: um processo incompleto é um processo que ou não é realizado ou é parcialmente realizado. Um ou mais objetivos específicos (SG) das PAs não são satisfeitos, e nenhum objetivo genérico existe para este nível, uma vez que não há razão para institucionalizar um processo realizado parcialmente.

1. Realizado: um processo realizado é aquele que satisfaz os SGs das PAs. Ele apóia e possibilita que o trabalho necessário para produzir os produtos de

trabalho seja realizado. Embora o nível de capacidade 1 resulte em importantes melhorias, elas podem ser perdidas com o tempo se elas não forem institucionalizadas. A aplicação da institucionalização (as práticas genéricas dos níveis de capacidade 2 a 5) auxiliam a assegurar que aquelas melhorias sejam mantidas.

2. *Gerenciado*: um processo gerenciado é um processo realizado (nível de capacidade 1) que tem uma infraestrutura básica para apoiar o processo. Ele é planejado e executado de acordo com uma política; emprega pessoas habilitadas que tem recursos adequados para produzir resultados controlados; envolve *stakeholders* relevantes; é monitorado, controlado e revisado; e é avaliado pela aderência à sua descrição. A disciplina de processos com nível de capacidade 2 auxilia a assegurar que práticas existentes sejam conservadas durante períodos de estresse.

3. *Definido*: um processo definido é um processo gerenciado (nível de capacidade 2) que é obtido a partir de um conjunto de processos padrão de acordo com as diretrizes da organização. A distinção entre os níveis de capacidade 2 e 3 é que o escopo de padrões, descrições de processos e procedimentos podem ser diferentes para cada instância específica do processo (para um projeto, por ex).

No nível de capacidade 2, os padrões, descrições de processo e procedimentos são diferentes para cada instância específica do processo (para um projeto particular, por ex.). Já no nível de capacidade 3, os padrões, descrições de processo e procedimentos para um projeto são adequados a partir de um conjunto de processos padrão e, portanto, são mais consistentes. Além disso, no nível de capacidade 3, os processos são tipicamente descritos de modo mais rigorosos do que no nível de capacidade 2.

Um processo definido define claramente o propósito, entradas, critérios de entrada, atividades, papéis, medidas, passos de verificação, saídas e critérios

de saída. No nível de capacidade 3, os processos são gerenciados mais proativamente usando um entendimento dos inter-relacionamentos entre as atividades do processo e medidas detalhadas do processo, seus produtos de trabalho e serviços.

4. Gerenciado Quantitativamente: um processo gerenciado quantitativamente é um processo definido (nível de capacidade 3) que é controlado usando técnicas estatísticas e outras quantitativas. Objetivos quantitativos para a qualidade e desempenho do processo são estabelecidos e usados como critérios para gerenciar o processo. A qualidade e o desempenho do processo é entendido em termos estatísticos e é gerenciado durante toda a vida do processo.

5. Em otimização: um processo em otimização é um processo gerenciado quantitativamente (nível de capacidade 4) que é melhorado com base no entendimento das causas comuns de variação no processo. O foco de um processo em otimização é melhorar continuamente o espectro do desempenho por meio de melhorias incrementais e inovadoras.

As regras para se alcançar determinado nível de maturidade são:

- Para alcançar o nível de maturidade 2, todas as PAs alinhadas ao nível de maturidade 2 devem atingir ao nível de capacidade 2 ou mais alto.
- Para alcançar o nível de maturidade 3, todas as PAs alinhadas aos níveis de maturidade 2 e 3 devem atingir ao nível de capacidade 3 ou mais alto.
- Para alcançar o nível de maturidade 4, todas as PAs alinhadas aos níveis de maturidade 2, 3 e 4 devem atingir ao nível de capacidade 3 ou mais alto.
- Para alcançar o nível de maturidade 5, todas as PAs devem atingir ao nível de capacidade 3 ou mais alto.

O nível de capacidade 4 para PAs não pode ser pré-determinado porque as escolhas dependem da seleção feita pela organização em implementar as PAs do nível de maturidade 4. O mesmo acontece para o nível de maturidade 5.

Todos os objetivos e práticas genéricas são usados na representação contínua. Na representação por estágios, somente os objetivos genéricos 2 e 3 são usados. Quando se tenta alcançar o nível 2, usa-se as PAs no nível de maturidade 2 e o objetivo genérico 2 com suas respectivas práticas. Os objetivos genéricos 4 e 5 e suas práticas genéricas associadas não são usados porque nem todos os processos serão elevados ao nível acima de um "processo definido". Somente os processos selecionados e sub-processos serão gerenciados quantitativamente e otimizados, e quais processos e sub-processos são selecionados é endereçado pelas PAs no nível de maturidade 4 e 5.

Quando se alcança os níveis de maturidade 3, 4 e 5, usa-se PAs nos níveis de maturidade apropriados e todas as PAs dos níveis de maturidade mais baixos. Em adição, o objetivo genérico 3 e suas práticas genéricas associadas (as quais incluem as práticas genéricas associadas com o objetivo genérico 2) são aplicadas para todas as PAs. Isso significa que se já foi alcançado o nível de maturidade 2, para alcançar o nível de maturidade 3 deve-se retornar para as PAs do nível de maturidade 2 e aplicar o objetivo genérico 3 e suas respectivas práticas.

Segundo Chrissis et al. (2003), três categorias de fatores podem influenciar a decisão sobre a escolha da arquitetura contínua ou por estágios: fatores de negócio, culturais e legados.

Curtis (1998) defende que os modelos contínuos são aqueles que desejam liberdade para escolher o foco da melhoria (atuando na melhoria de cada processo em separado) e que os modelos por estágios são aqueles que desejam a mudança da cultura da organização.

Já para Kasse¹⁸ (2004) apud (SALVIANO, 2006), a arquitetura por estágios auxilia a organização a priorizar seus esforços de melhoria, especialmente quando a organização é principiante na área de melhoria de processo ou está em um nível baixo de maturidade e tem pouca experiência em estabelecimento de processo. E, a

¹⁸ KASSE, T. (2004). Practical Insight into CMMI. Artech House Publishers.

arquitetura contínua do CMMI auxilia a organização a construir um perfil alvo de áreas de processo que auxiliará a organização a resolver objetivos estratégicos conhecidos.

Olson¹⁹ (2003) apud (SALVIANO, 2006) diz que a questão não é qual representação, por estágios ou contínua, deve ser usada. Primeiramente, devem ser identificadas as metas de qualidade, objetivos e estratégia da organização para depois definir uma estratégia de melhoria: melhoria da qualidade, controle da qualidade ou planejamento da qualidade. Por exemplo: (1) se uma organização tem o orçamento limitado, precisa economizar recursos e ter um retorno de investimento rápido, então deve escolher uma estratégia de melhoria da qualidade, o que sugere a arquitetura contínua com a escolha de uma área de processo para orientar a melhoria; (2) se uma organização deseja ser a melhor dentre o seu nicho de mercado em um prazo longo, e deseja chegar nesta posição de forma ordenada, pode selecionar a estratégia de planejamento da qualidade, o que sugere a arquitetura por estágios.

Embora existam muitas razões para usar a representação contínua, os índices fornecidos pelos perfis de capacidade de processo²⁰ são limitados em sua habilidade de fornecer às organizações um caminho para compará-la, de modo geral, com outras organizações. Os perfis de capacidade podem ser usados, nesse caso, se cada organização seleciona as mesmas PAs; de qualquer maneira, os níveis de maturidade tem sido usados para comparar as organizações ao longo dos anos e já fornece um conjunto pré-definido de PAs. Por isso, foi criada uma equivalência entre perfis de níveis de capacidade e níveis de maturidade. Assim, uma organização que usa a representação contínua para uma avaliação, pode converter o perfil de capacidade de processo a um nível de maturidade associado (SOFTWARE ENGINEERING INSTITUTE, 2006).

¹⁹ OLSON, T. G. (2003). **Staged or Continuous: Which Model Should I Choose?**, Slides from presentation at Third Annual CMMI Technology Conference and Users Group, USA, November 2003.

²⁰ Um **perfil de capacidade** é uma lista de PAs e os níveis de capacidade correspondentes para cada. O perfil possibilita que uma organização rastreie seu nível de capacidade em relação às PAs. O perfil é denominado perfil realizado quando representa o progresso atual de cada PA. Alternativamente, o perfil é um perfil almejado quando representa objetivos planejados de melhoria dos processos da organização.

3.4.3. MPS.BR e Modelo MR-MPS

A Associação para Promoção da Excelência do Software Brasileiro (SOFTEX) iniciou um programa mobilizador denominado MPS.BR (Melhoria de Processos do Software Brasileiro) com o intuito de melhorar continuamente a qualidade de software no Brasil a um custo acessível. Isso devido à dificuldade e alto custo de implantação e avaliação do CMMI (SOFTWARE ENGINEERING INSTITUTE, 2002) e a necessidade de expandir o mercado de software brasileiro (WEBER et al., 2005b).

O Modelo MPS.BR teve como bases de construção as normas ISO/IEC 12207 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2001) e ISO/IEC 15504 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2003) e as melhores práticas da engenharia de software, e mantém compatibilidade com o CMMI.

A sua originalidade não está no conteúdo, mas sim na estratégia de implementação, criada para a realidade brasileira (SOFTEX, 2006). Esse modelo possui três componentes:

- **Modelo de Referência (MR-MPS):** é definido por meio de níveis de maturidade, seqüenciais e acumulativos. Cada nível de maturidade é uma junção entre processos e capacidade dos processos, ou seja, é composto por um conjunto de processos em um determinado nível de capacidade. O progresso e o atendimento do nível de maturidade se obtém quando são atendidos todos os resultados e propósito do processo; e os atributos de processo relacionados aquele nível.
- **Método de Avaliação (MA-MPS):** descreve o modo como a avaliação deve ser efetuada, quais os requisitos dos avaliadores e os requisitos para atender ao MR-MPS. O MA-MPS foi definido em conformidade com os requisitos para modelos de referência de processo e métodos de avaliação de processos estabelecidos na norma ISO/IEC 15504-2 e atende aos requisitos específicos do Programa MPS.BR. Assim, o método está em conformidade com a ISO/IEC 15504 e é compatível com o método SCAMPI

para avaliação, segundo o modelo CMMI, também definido com base na ISO/IEC 15504.

- **Modelo de Negócio (MN-MPS):** compreende (i) um Modelo de Negócio Cooperado (MNC-MPS) - próprio para grupos de pequenas e médias empresas que necessitam melhorar radicalmente seus processos de software; (ii) um Modelo de Negócio Específico (MNE-MPS) - próprio para empresas de qualquer porte e natureza que não querem compartilhar com outras empresas a melhoria de seus processos de software.

3.4.4. ISO 9000

A série ISO 9000 foi elaborada através de um consenso internacional sobre as práticas que uma empresa pode tomar a fim de atender plenamente os requisitos de qualidade do cliente. A ISO 9000 não fixa metas a serem atingidas pelas empresas a serem certificadas, a própria empresa é quem estabelece as metas a serem atingidas.

As normas da família ISO 9000-2000 foram desenvolvidas para apoiar organizações, de todos os tipos e tamanhos, na implementação e operação de sistemas de gestão da qualidade eficazes.

ISO 9000 inclui as seguintes normas: ISO 9000:2000, ISO 9001:2000 e ISO 9004:2000. A ISO 9000 descreve os fundamentos de sistemas de gestão da qualidade e estabelece a terminologia para estes sistemas. A ISO 9001 (INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2000) especifica requisitos para um sistema de gestão da qualidade, onde uma organização precisa demonstrar sua capacidade para fornecer produtos que atendam os requisitos do cliente e os requisitos regulamentares aplicáveis, e objetiva aumentar a satisfação do cliente. A ISO 9004 fornece diretrizes que consideram tanto a eficácia como a eficiência do sistema de gestão da qualidade. O objetivo desta norma é melhorar o desempenho da organização e a satisfação dos clientes e das outras partes interessadas.

3.4.5. PMBOK

O Corpo de Conhecimento da Gestão de Projeto (PMBOK – *Project Management Body of Knowledge*) foi desenvolvido pelo *Project Management Institute* (PMI) para ser um guia que disponibiliza conhecimento relacionado à gestão de projetos.

Contém um conjunto de práticas já sedimentadas em gestão de projetos, coletadas por profissionais envolvidos em projetos de todo o mundo. E, passa por evoluções de 4 em 4 anos, baseadas em contribuições de seus membros. É aplicável em qualquer área inclusive em projetos de software.

De acordo com o PMBOK (PROJECT MANAGEMENT INSTITUTE, 2004), a gestão de projetos ocorrerá em áreas de conhecimento, por meio da aplicação e integração de processos. As áreas de conhecimento totalizam nove: escopo do projeto, tempo, custo, qualidade, recursos humanos, comunicações, riscos, aquisições e integração do projeto.

As fases da gestão de projeto que cobrem as áreas de conhecimento com maior ou menor ênfase são: *Iniciação*, onde é definido e autorizado o projeto ou fase do projeto; *Planejamento*, onde são definidos e refinados os objetivos, planejada a ação necessária para alcançar os objetivos e o escopo do projeto; *Execução*, onde há a integração das pessoas e outros recursos para a realização do plano de gestão do projeto; *Monitoramento e Controle* do progresso do projeto para identificar variações em relação ao plano de gestão do projeto, de forma que possam ser tomadas ações corretivas quando necessário para atender aos objetivos do projeto; *Encerramento*, onde é formalizado a aceitação do produto, serviço ou resultado, conduzindo ao final do projeto ou de fase do projeto.

3.4.6. SWEBOK

O Corpo de Conhecimento de Engenharia de Software (*SWEBOK – Software Engineering Body of Knowledge*) é coordenado por um projeto da *IEEE Computer Society* e *ACM (Association for Computing Machinery)* em busca dos seguintes objetivos: promover uma visão consistente mundialmente da engenharia de software, caracterizar o conteúdo da engenharia de software, esclarecer o escopo e limites da engenharia de software em relação a outras disciplinas (como engenharia da computação, ciência da computação, matemática, gestão de projetos, gestão da qualidade, ergonomia de software e engenharia de sistemas) e prover uma fundação para o desenvolvimento de currículos e material para certificações individuais (ABRAN et al., 2004).

O primeiro dos objetivos anteriormente citados – promover uma visão consistente mundialmente da engenharia de software – foi suportado por um processo de desenvolvimento no qual estavam engajados, aproximadamente, 500 revisores de 42 países na elaboração da versão experimental (1998-2001) e, depois, mais 120 revisores de 21 países no ano de 2003 na elaboração da versão 2004.

3.5. Considerações Finais

Há na literatura vários modelos de melhoria de processo, como os apresentados neste capítulo, que podem ser utilizados como referência para compor o contexto de utilização de um modelo de processo de software.

Os *frameworks* para modelos de capacidade de processo²¹ e métodos de avaliação de processo - CMMI e ISO/IEC 15504 - são os mais relevantes no setor de software (SALVIANO, 2006). O MPS.BR é compatível com o CMMI.

²¹ Alguns modelos utilizam níveis de “maturidade”, tais como: SW-CMM, CMMI-SE/SW e CMMI-DEV. Essa maturidade é definida em função da capacidade de processo, por isso, o termo “capacidade” é suficiente para caracterizar esses modelos, também.

4. METODOLOGIA DE PESQUISA

4.1. Considerações Iniciais

A pesquisa é uma atividade voltada para a solução de problemas teóricos ou práticos com o emprego de processos científicos, ou seja, ela parte de uma dúvida ou problema, e com o uso do método científico, busca uma resposta ou solução (CERVO & BERVIAN, 2002).

Método científico é o conjunto de processos ou operações mentais que se devem empregar na investigação. É a linha de raciocínio adotada no processo de pesquisa.

Para o estabelecimento do método científico é necessário definir o referencial teórico, isto é, a categoria de métodos de pesquisa mais amplos - aqueles diretamente relacionados com as correntes que estudam métodos de pesquisa dentro da área de epistemologia (PÁDUA, 1996). Os referenciais teóricos existentes são: dedutivo, indutivo, dialético, fenomenológico e hipotético-dedutivo (GIL, 1999; MARCONI & LAKATOS, 2000).

Na literatura, há diversas classificações de tipos de pesquisas, sendo que os critérios para essas classificações variam de acordo com o enfoque dado (BRYMAN, 1989; DANE, 1990; PÁDUA, 1996; GIL, 1999; CERVO & BERVIAN, 2002; MARCONI & LAKATOS, 2000). Há classificações do ponto de vista da natureza da pesquisa, dos objetivos de pesquisa e dos procedimentos técnicos da pesquisa.

Neste capítulo será apresentado o referencial teórico que embasou a pesquisa realizada neste trabalho e as classificações da pesquisa, de acordo com os vários critérios existentes.

4.2. Abordagens e métodos de pesquisa

O referencial teórico adotado neste trabalho é o **hipotético-dedutivo**, que se inicia pela percepção de uma lacuna nos conhecimentos, acerca da qual se formula hipóteses e, pelo processo de inferência dedutiva, testa-se a predição da ocorrência de fenômenos abrangidos pela hipótese. Desse modo, a busca da solução do problema é feita quando teorias ou leis falham na solução de um problema (refutação) e, então, é proposta uma nova teoria ou lei (conjectura) que resolva o problema e incorpore a teoria ou lei anterior. Assim, cada vez que a teoria resistir a um teste para refutação será considerada mais robusta (POPPER²², 1975 apud MARCONI & LAKATOS, 2000).

Tal referencial teórico é adequado a este trabalho de pesquisa, uma vez que a análise do Modelo elaborado foi realizada por meio da tentativa de refutação de uma hipótese, durante a implantação das atividades do modelo em ambientes reais de MPEs e durante as análises críticas de profissionais da área.

O presente trabalho considerou a seguinte hipótese: *“É possível criar um modelo que considere tanto um modelo de processo de software quanto modelos de capacidade de processo em um contexto de melhoria de processo, a fim de orientar a definição por parte de uma MPE de seu processo padrão de venda e desenvolvimento de software on-demand, de maneira mais detalhada do que a sugerida pelos modelos de capacidade de processo”*.

Do ponto de vista da natureza da pesquisa, ela pode ser classificada em *pesquisa pura ou básica* e *pesquisa aplicada* (CERVO & BERVIAN, 2002). Nesse aspecto, este trabalho de pesquisa é classificado como uma pesquisa aplicada, uma vez que objetiva gerar conhecimentos para aplicação prática dirigida à solução de um problema específico: auxiliar as MPEs na definição e melhoria de seu processo de

²² POPPER, K. S. (1975). A lógica da pesquisa científica. 2ª ed. São Paulo: Cultrix.

venda e desenvolvimento de software, em conformidade com os modelos de capacidade de processo CMMI e ISO/IEC 15504.

Do ponto de vista da forma de abordagem do problema, a classificação de pesquisa pode ser: *quantitativa* e *qualitativa* (BRYMAN, 1989). Considerando que este trabalho de pesquisa não buscou provar estatisticamente os benefícios e adequação da integração do modelo de processo de software aos modelos de capacidade de processo, pode ser caracterizado do ponto de vista da forma de abordagem do problema como uma abordagem qualitativa.

Do ponto de vista dos objetivos de pesquisa, ela pode ser classificada, segundo GIL (1999), como *exploratória*, *descritiva* e *explicativa*. Para Dane (1990), além dessas há a *pesquisa preditiva* e a *pesquisa ação*.

De acordo com o objetivo da pesquisa esse trabalho pode ser classificado como **pesquisa-ação**. Segundo Thiollent (2002), na pesquisa-ação os pesquisadores desempenham um papel ativo no equacionamento dos problemas encontrados, no acompanhamento e na avaliação das ações desencadeadas em função dos problemas. Sendo assim, neste tipo de pesquisa, além da exigência do envolvimento ativo do pesquisador e ação por parte das pessoas ou dos grupos envolvidos no problema, é necessário produzir conhecimentos, adquirir experiência, contribuir para a discussão da área estudada ou fazer avançar o debate acerca das questões abordadas.

Neste trabalho de pesquisa, o pesquisador agiu na busca de uma solução para orientar mais detalhadamente às MPEs durante a definição e melhoria de seu processo de venda e desenvolvimento de software *on-demand*. Isso foi realizado aplicando o modelo elaborado neste trabalho de pesquisa em duas MPEs, aprendendo com os resultados, obtendo críticas e sugestões de profissionais da área.

Do ponto de vista dos procedimentos técnicos, as principais classificações são (PÁDUA, 1996; CERVO & BERVIAN, 1983): pesquisa bibliográfica, pesquisa documental, pesquisa experimental, levantamento, pesquisa de campo e estudo de

caso. Nesse trabalho de pesquisa foram utilizadas: a **pesquisa bibliográfica** para o levantamento dos assuntos abordados no trabalho de pesquisas.

Dentre o conjunto de técnicas de coleta de dados existente na literatura (MARCONI & LAKATOS, 1999), nesse trabalho de pesquisa foram utilizados: 1) um questionário (Apêndice 1) aplicado a 5 (cinco) MPE's que desenvolvem software e 9 (nove) CPDs (Centro de Processamento de Dados) de empresas diversas os quais desenvolvem software para uso interno, que confirmou a seleção das áreas de conhecimento definidas no Modelo ProcSoftVD e indicou a importância de outras áreas de conhecimento que serão consideradas em um trabalho futuro; 2) um questionário (Apêndice 7) entregue a profissionais que trabalham em MPEs ou já trabalharam em empresas e também profissionais com mestrado e doutorado na área, para levantarem os pontos fortes e fracos do modelo e para estabelecerem o nível de atendimento do modelo em relação às características de qualidade: funcionalidade, usabilidade, manutenibilidade, portabilidade e eficiência; 3) entrevistas com esses profissionais a fim de obter sugestões para melhorar o modelo.

4.3. Etapas do trabalho

Primeiramente, foi realizada a revisão bibliográfica para fundamentar a proposta elaborada nesse trabalho de pesquisa. Os assuntos pesquisados foram: modelos de processo de software, modelos de capacidade de processo e modelos/métodos/abordagens de melhoria de processo.

As outras atividades realizadas neste trabalho de pesquisa foram agrupadas em três etapas (Figura 4.1):

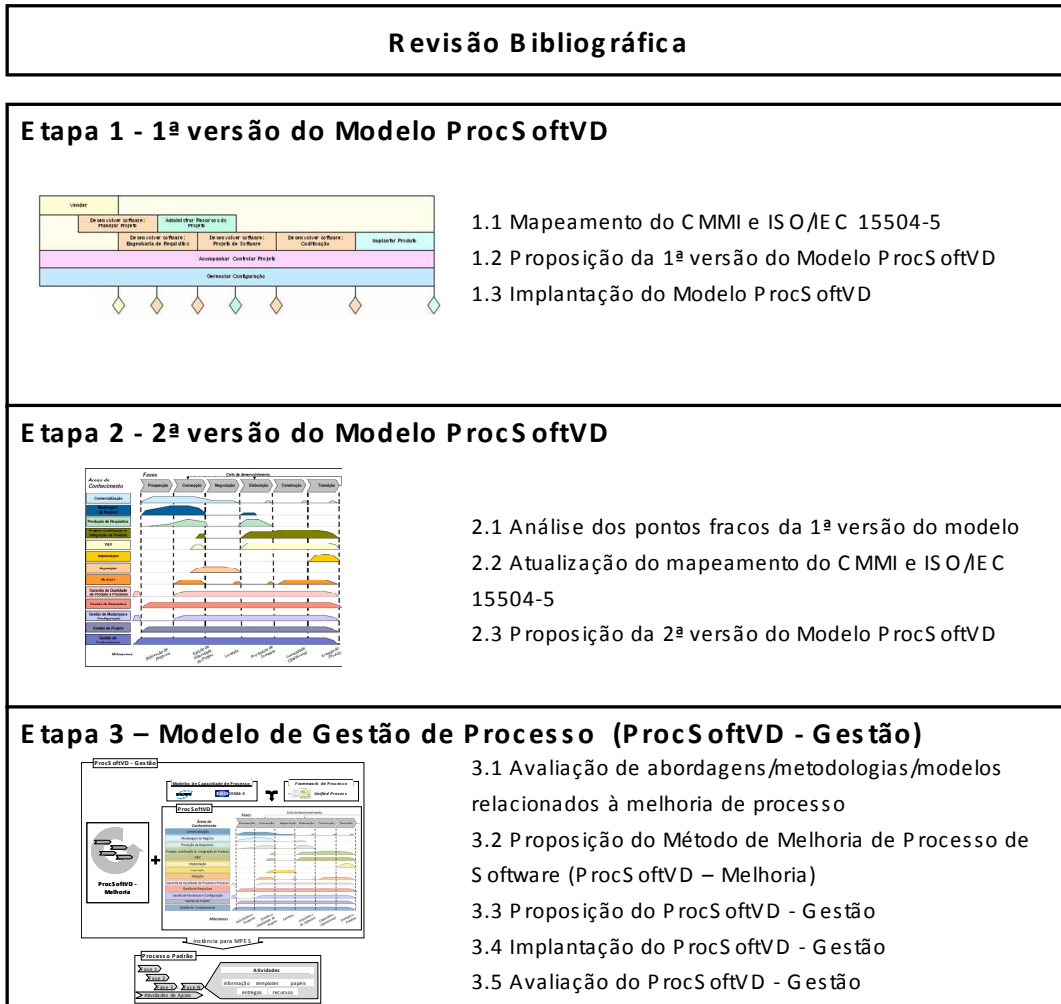


FIGURA 4.1 – ETAPAS DO TRABALHO DE PESQUISA

- **Etapa 1:** a principal entrega dessa etapa foi uma primeira versão do modelo ProcSoftVD (seção 5.2). Para isso, foram realizadas as seguintes atividades:
 - 1.1 *Mapeamento do CMMI e ISO/IEC 15504-5:* foi mapeada a equivalência dos objetivos específicos do nível de maturidade 2 do CMMI em relação às práticas-base da parte 5 do relatório técnico da ISO/IEC 15504 para, posteriormente, associar cada uma das atividades do modelo de processo proposto (item 1.2) a esses objetivos específicos e práticas-base. Essa associação das atividades do modelo ProcSoftVD com o

CMMI e ISO/IEC 15504-5 serve para mostrar ao usuário do modelo, caso seja necessário, que as atividades do modelo ProcSoftVD atendem às melhores práticas sugeridas pelo CMMI e ISO/IEC 15504-5; que o modelo ProcSoftVD está em consonância com modelos/normas internacionais de qualidade de processo.

- 1.2 *Proposição da 1ª versão do Modelo ProcSoftVD*: essa versão do modelo foi elaborada com base na estrutura do modelo de processo cascata; uma estrutura seqüencial e linear que dificilmente retrata a maioria dos projetos de desenvolvimento de software *on-demand*. Foram definidas as atividades de cada uma das fases, suas tarefas, artefatos de entrada e saída, papéis dos responsáveis pela execução das atividades, recursos a serem utilizados e *templates* (formulários com uma estrutura definida para cada documento a ser criado pelas atividades do modelo). Foram elaborados: um pôster contendo a visão gráfica de todas as atividades de cada uma das fases do modelo (Figura 5.2); um pôster contendo a visão gráfica dos documentos a serem utilizados/criados nas/pelas atividades do modelo (Figuras 5.5 e 5.6); um manual contendo o detalhamento de cada uma das atividades das fases do modelo (Figura 5.7) e um manual contendo a estrutura de cada um dos documentos a serem utilizados/criados nas/pelas atividades do modelo (Figura 5.8).
- 1.3 *Implantação do Modelo ProcSoftVD*: o modelo definido anteriormente foi instanciado para a Empresa 1 (Figura 5.3) e, a partir dessa instância, por meio da pesquisa-ação foram implantadas duas fases - fases Vender e Planejar Projeto.

- **Etapa 2:** a principal entrega dessa etapa foi uma segunda versão do modelo ProcSoftVD (seção 5.2). Para isso, foram realizadas as seguintes atividades:

2.1 *Análises dos pontos fracos da 1ª versão do modelo:* durante a pesquisa-ação realizada na Empresa 1, os funcionários da empresa junto ao pesquisador deste trabalho detectaram que a estrutura do modelo (seqüencial e linear) não era adequada para o projeto piloto o qual foi utilizado durante a implantação das fases Vender e Planejar Projeto. Percebeu-se, assim, a necessidade de alteração na estrutura do modelo ProcSoftVD para atender a maioria dos projetos de desenvolvimento de software *on-demand*. Também foram detectados alguns pontos fracos em algumas atividades do modelo que deveriam ser melhorados em uma próxima versão do modelo.

2.2 *Atualização do mapeamento do CMMI e ISO/IEC 15504-5:* foi publicada uma versão mais atual do CMMI – CMMI-DEV v1.2 e foi obtida a norma ISO/IEC 15504. Por isso, foi realizada a atualização do mapeamento.

2.3 *Proposição da 2ª versão do Modelo ProcSoftVD:* uma vez identificada a deficiência da estrutura do modelo (item 2.1), a estrutura do modelo foi modificada para aceitar projetos de desenvolvimento de software mais condizentes à realidade: feitos de forma incremental e evolutiva. Foram também definidos, em meio à pesquisa-ação, os requisitos (seção 5.2.1) a serem atendidos pelo modelo ProcSoftVD; foram re-elaborados os elementos do modelo: atividades, tarefas, artefatos de entrada e saída, papéis, recursos e *templates* para cada atividade do modelo ProcSoftVD e para facilitar o entendimento da associação dos elementos do modelo ProcSoftVD foi elaborado um meta-modelo (seção 5.2.2).

- **Etapa 3:** a principal entrega dessa etapa foi o Modelo de Gestão do Processo de Venda e Desenvolvimento de Software *On-demand* para MPEs (ProcSoftVD - Gestão) – a integração do modelo ProcSoftVD com um método de melhoria de processo (capítulo 5). Para isso, foram realizadas as seguintes atividades:

3.1 *Avaliação de abordagens, metodologias e modelos relacionados à melhoria de processo:* para a implantação de algumas fases da 1ª versão do modelo ProcSoftVD na Empresa 1, foi utilizada a Metodologia de Gestão de Mudanças elaborada por Costa (2006). Questionou-se sobre a existência na literatura de outras metodologias/abordagens/métodos voltadas especificamente para melhoria de processo de software para MPEs. Foi realizado, então, um levantamento bibliográfico sobre abordagens de melhoria de processo existentes na literatura e, posteriormente, foi realizada uma seleção, entre as abordagens de melhoria de processo levantadas, das mais adequadas para serem utilizadas pelas MPE's de software.

3.2 *Proposição do Método de Melhoria de Processo de Software (ProcSoftVD – Melhoria)* (seção 5.3): a partir da avaliação das abordagens, metodologias e modelos relacionados à melhoria de processo (item 3.1) foi identificada a complementaridade entre algumas delas e, com base nessa complementaridade foi elaborado um método de melhoria de processo (ProcSoftVD – Melhoria).

3.3 *Proposição do ProcSoftVD – Gestão* (capítulo 5): considerando o contexto de gestão de processos de negócio, mais especificamente o de melhoria, foi elaborado o ProcSoftVD – Gestão (Figura 5.1), um Modelo de Gestão do Processo de Venda e Desenvolvimento de Software *On-demand* que tem o objetivo de orientar de maneira

detalhada às MPEs a definirem o seu processo padrão de venda e desenvolvimento de software *on-demand*.

3.4 *Implantação do ProcSoftVD - Gestão* (seção 6.2): foi aplicado o Método de Melhoria de Processo (ProcSoftVD – Melhoria), elaborado anteriormente, em uma MPE (Empresa 2) (Apêndice 6). Nessa aplicação foram definidos ciclos de melhoria compostos por áreas de conhecimento a serem implantadas, de acordo com a prioridade da empresa. Assim, foi realizada a implantação das áreas de conhecimento “Modelagem de Negócios” e “Produção de Requisitos” do Modelo de Processo de Venda e Desenvolvimento de Software *On-demand* para MPEs (ProcSoftVD).

3.5 *Avaliação do ProcSoftVD - Gestão* (seção 6.3): foram realizados questionários (Apêndice 7) e entrevistas com 6 (seis) profissionais da área de computação, sendo que 5 (cinco) deles tem experiência como analista de sistemas/programador e 4 (quatro) deles tem mestrado/doutorado na área de computação e atuantes como docente em universidades.

4.4. Considerações Finais

Neste capítulo foram apresentados o referencial teórico que embasou a pesquisa realizada neste trabalho, as classificações da pesquisa, de acordo com os vários critérios existentes e as etapas do trabalho de pesquisa.

No próximo capítulo será apresentado a principal entrega resultante deste trabalho de pesquisa – o Modelo ProcSoftVD – Gestão.

5. Modelo de Gestão do Processo de Venda e Desenvolvimento de Software *On-demand* para MPEs (ProcSoftVD – Gestão)

5.1. Considerações Iniciais

Neste capítulo é apresentado o ProcSoftVD - Gestão (Figura 5.1) - um modelo de Gestão do Processo de Venda e Desenvolvimento de software *on-demand* para MPEs, elaborado a partir da integração do Modelo do Processo de Venda e Desenvolvimento de software *on-demand* para MPEs (ProcSoftVD) (seção 5.2) com o Método de Melhoria do Processo de Software (ProcSoftVD – Melhoria) (seção 5.3).

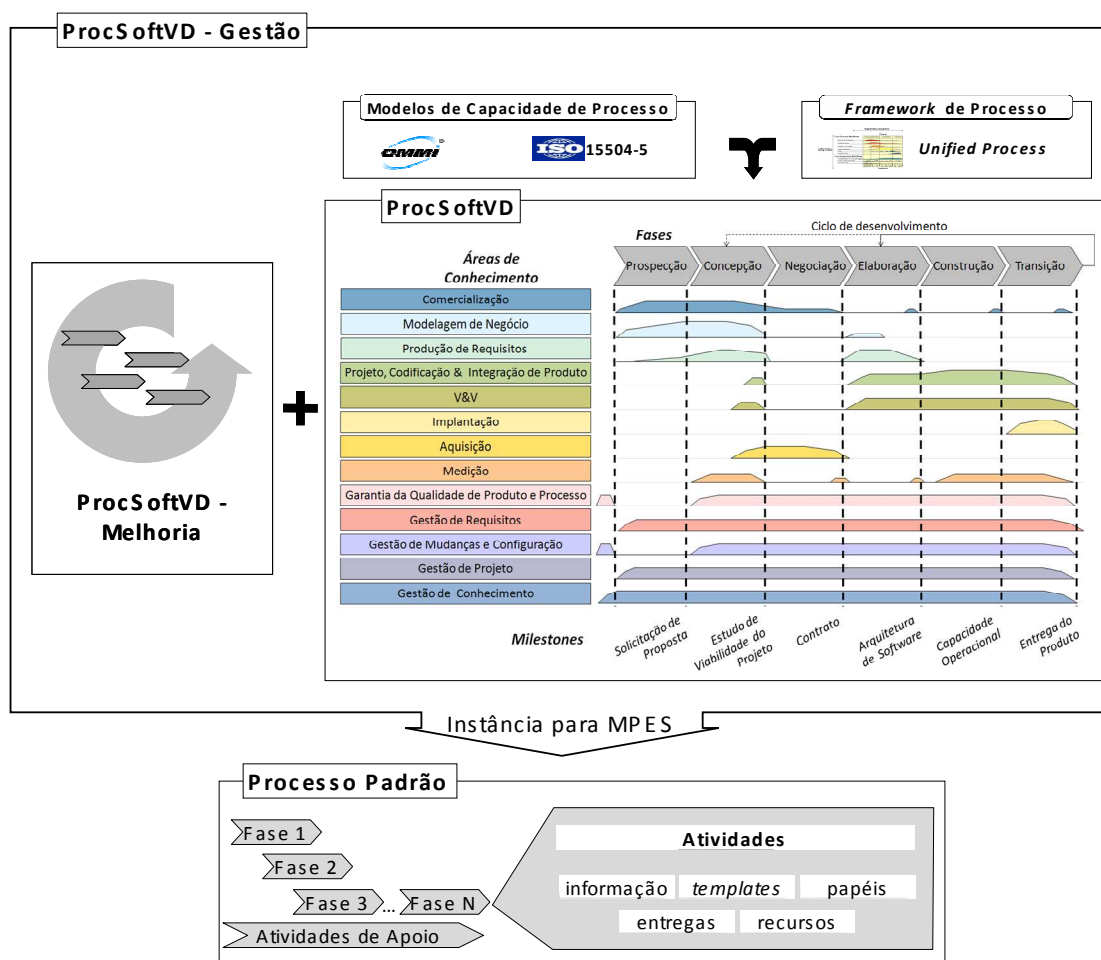


FIGURA 5.1 – MODELO DE GESTÃO DO PROCESSO DE VENDA E DESENVOLVIMENTO DE SOFTWARE ON-DEMAND PARA MPEs (PROCSoftVD - GESTÃO)

Além disso, na seção 5.4, é apresentada a publicação *web* do modelo ProcSoftVD – Gestão que pode ser acessada por qualquer interessado.

5.2. Evolução do Modelo de Processo de Venda e Desenvolvimento de Software *On-demand* para MPEs (ProcSoftVD)

Inicialmente, esse projeto de pesquisa tinha como objetivo principal melhorar o desempenho de empresas que pertenciam a um APL (arranjo produtivo local) (CASSIOLATO & LASTRES, 2001); participavam do projeto duas empresas. Para isso, foram definidos alguns objetivos secundários, dentre eles o de modelar um Processo Genérico para Venda e Desenvolvimento de Software (PV&DS) *On-Demand* Colaborativo. Um dos resultados parciais desse projeto foi uma versão inicial do Modelo Genérico do PV&DS *on-demand* (Figura 5.2), elaborada a partir do mapeamento das práticas do nível de maturidade 2 do CMMI v1.1 com as práticas definidas no modelo ISO/IEC TR 15504-1998.

Entretanto, depois de um ano de andamento do projeto anteriormente referido, devido a mudanças estratégicas de uma das empresas do APL, apenas a Empresa 1 continuou no projeto. Trata-se de uma pequena empresa de base tecnológica, fundada em janeiro de 2000, cujo alvo é o desenvolvimento de software, mais especificamente o desenvolvimento de sistemas para Internet, *e-learning* e software sob encomenda.

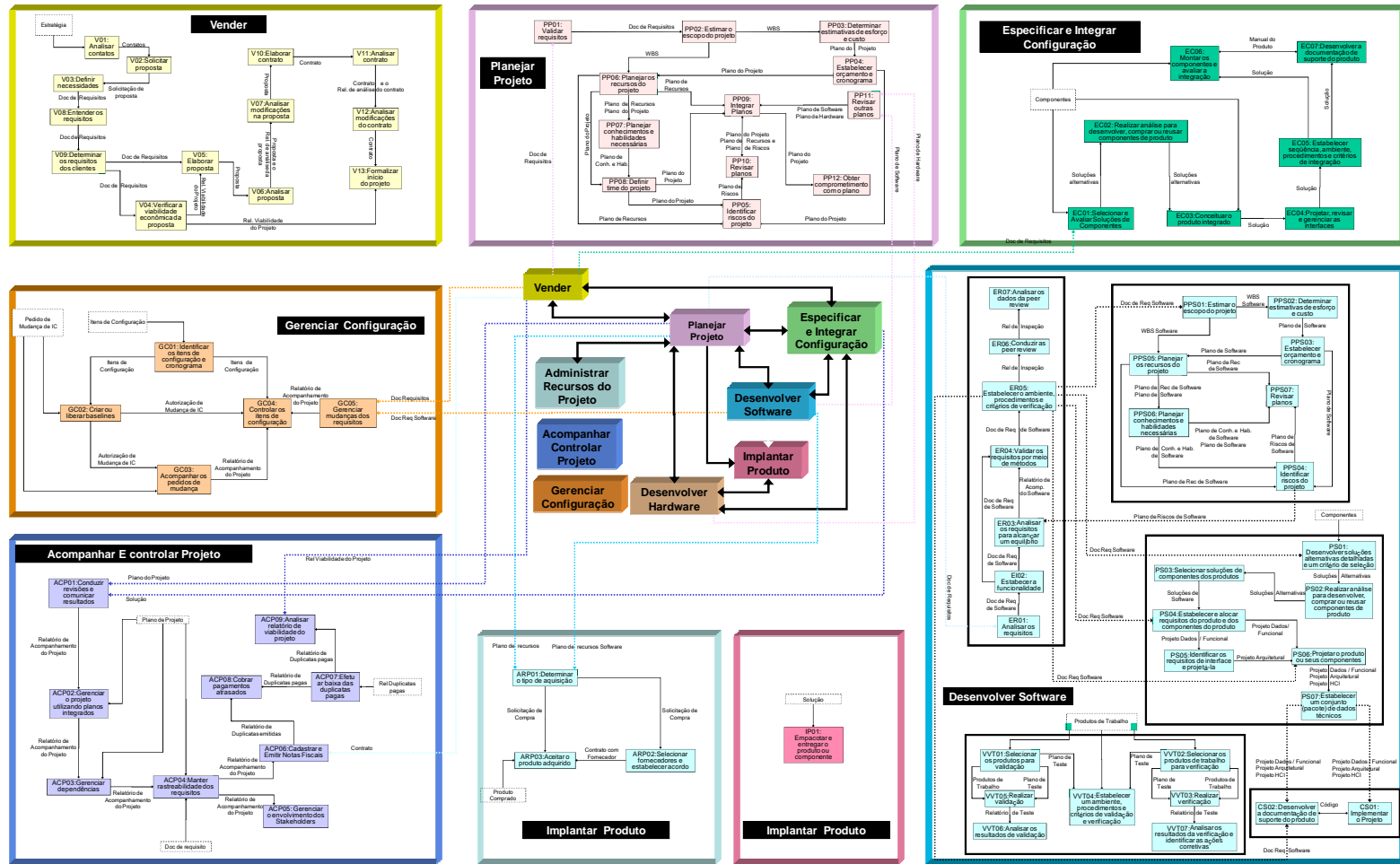


FIGURA 5.2 – PRIMEIRA VERSÃO DO MODELO GÊNÉRICO DO PV&DS ON-DEMAND

Aplicando a Metodologia de Gestão de Mudanças proposta em COSTA (2006) na Empresa 1, foi revelado que não existiam padrões de documentos e nem uma visão sistemática de suas atividades, implicando no desenvolvimento de projetos de sistemas com base na experiência do gerente de projeto em conduzir esses projetos. Além disso, havia uma grande rotatividade de pessoal na empresa dificultando o desenvolvimento de sistemas a contento de ambas as partes envolvidas – cliente e empresa, no prazo estabelecido.

Foi, então, definido um portfólio de projetos de mudança para a Empresa 1, tendo como primeiro projeto de mudança a instanciação do Modelo Genérico do PV&DS *on-demand* para a Empresa 1. As entregas dessa instanciação foram: um pôster com a visão geral das atividades do modelo proposto, agrupadas em fases, com seus respectivos fluxos de entradas e saídas (Figura 5.3; nessa figura estão destacadas em vermelho pontilhado as atividades da fase “Vender” que podem ser melhor visualizadas na Figura 5.4); um pôster do conteúdo dos documentos do modelo instanciado para facilitar o entendimento comum a todos (Figura 5.5; nessa figura está destacado em vermelho pontilhado o artefato “Documento de Requisitos” que pode ser melhor visualizado na Figura 5.6); um manual contendo o detalhamento de todas as atividades do modelo de processo proposto (Figura 5.7) e um manual com os *templates* (modelos de documentos) (Figura 5.8) para os artefatos utilizados/criados nas/por meio das atividades do modelo de processo proposto.

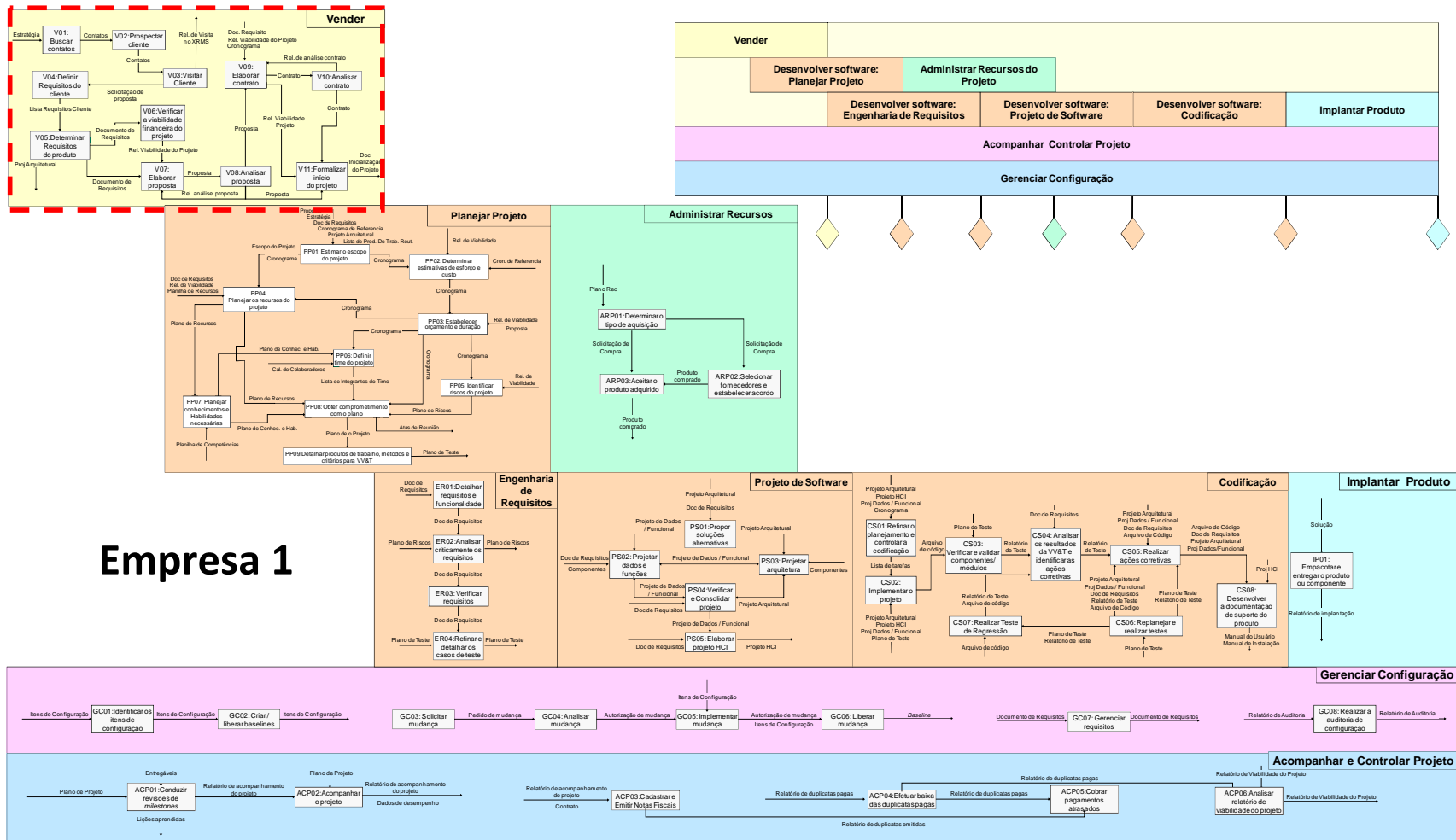


FIGURA 5.3 – INSTÂNCIA DO MODELO GENÉRICO DO PV&DS ON-DEMAND PARA A EMPRESA 1

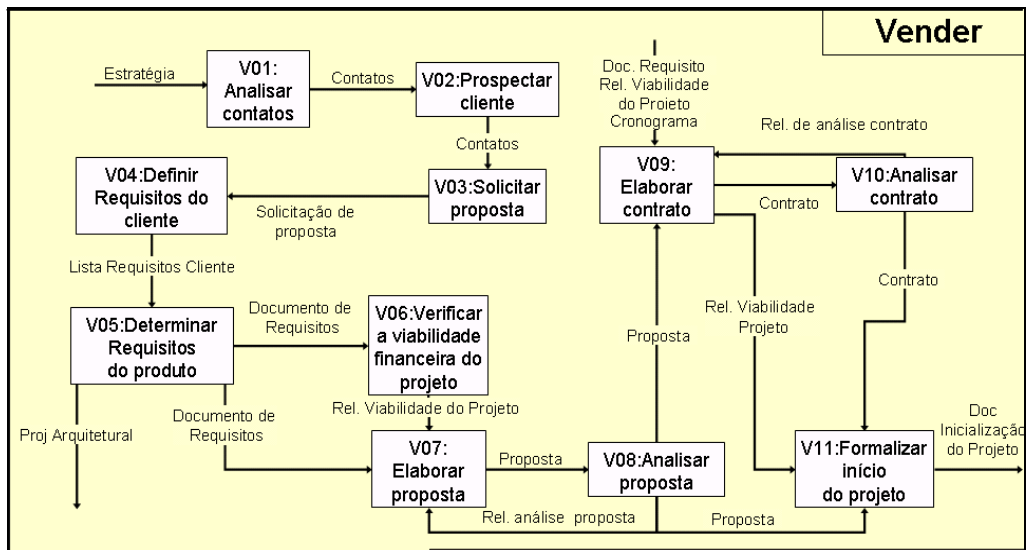


FIGURA 5.4 - VISÃO GRÁFICA DA FASE VENDER

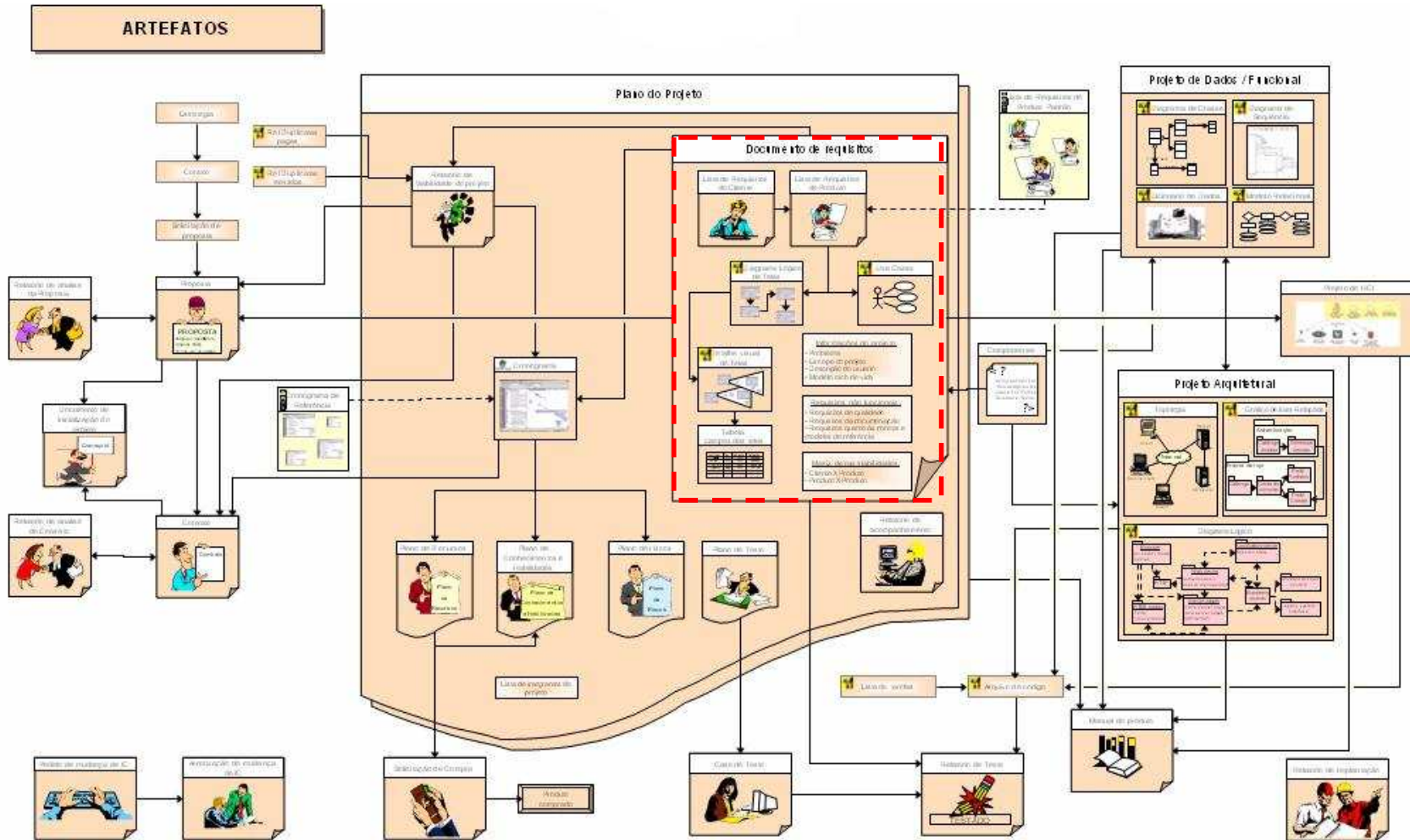


FIGURA 5.5- VISUALIZAÇÃO GRÁFICA DO CONTEÚDO DOS ARTEFATOS DO PROCESSO PROPOSTO

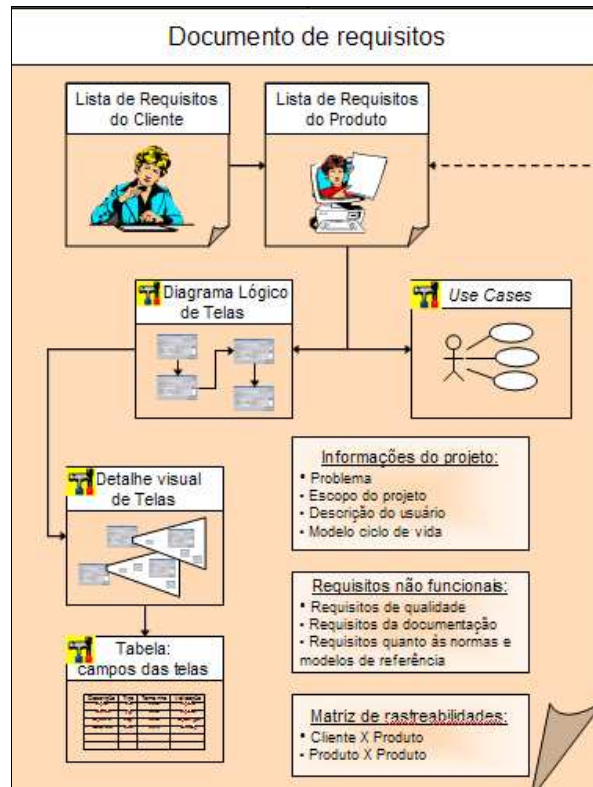


FIGURA 5.6 - VISUALIZAÇÃO GRÁFICA DO CONTEÚDO DO ARTEFATO “DOCUMENTO DE REQUISITOS”

Atividade	V05. Determinar requisitos do produto
Objetivo	Transformar os requisitos do cliente em requisitos do produto, uma vez que uma necessidade do cliente pode ser traduzida em mais de um requisito do produto; a descrição dos requisitos dependerá do nível de detalhamento definido. Também, quando necessário, já devem ser elaboradas as telas do produto
Tarefas	<ul style="list-style-type: none"> • Definir o ID do projeto • Selecionar topologia do sistema • Selecionar uma lista de requisitos do produto padrão • Traduzir a lista de requisitos do cliente em requisitos do produto descrevendo os nomes, tipos, descrição e prioridade dos requisitos • Marcar quais dos requisitos serão colocados na proposta • Marcar quais dos requisitos serão colocados no contrato • Definir o nível de detalhamento do Documento de Requisitos e Projeto Arquitetural nesta fase • Criar os requisitos definidos conforme o nível de detalhamento
Papéis	Gerente de Projeto Equipe de Vendas
Entradas	Lista de requisitos do cliente
Saídas	Documento de Requisitos (T)* (inclui a Lista de Requisitos do Produto (T)) Projeto Arquitetural (T)*
Recursos	

Atividade	V06. Verificar a viabilidade financeira do projeto
Objetivo	Avaliar, com base no Documento de Requisitos e na base de dados histórica, se é viável a execução do projeto em termos financeiros
Tarefas	<ul style="list-style-type: none"> • Definir <i>cost driver</i> • Definir valor da unidade do <i>cost driver</i> • Definir o custo operacional geral produto • Estimar valores de possíveis aquisições • Definir premissas financeiras • Definir o valor factível • Fazer o fluxo de caixa do projeto • Definir os indicadores financeiros do projeto • Definir o cronograma de recebimentos • Definir o preço
Papéis	Gerente do Projeto Equipe de Vendas
Entradas	Documento de Requisitos (T)
Saídas	Relatório de Viabilidade do Projeto (T)
Recursos	

FIGURA 5.7 - DETALHAMENTO DE UMA ATIVIDADE DO MODELO DE PROCESSO PROPOSTO

25 RELATÓRIO DE TESTE

ID documento: _____ Data: / / _____ Versão : _____

Responsável pelo documento: _____

Técnico responsável: _____

ID Projeto: _____

ITENS TESTADOS

PROCEDIMENTOS APLICADOS NO TESTE

AÇÕES CORRETIVAS

Descrição das ações corretivas	Status

Manual de Templates
45/50

FIGURA 5.8 – EXEMPLO DE TEMPLATE

Após uma análise crítica em relação às entregas, citadas anteriormente, por parte dos colaboradores da Empresa 1, foi iniciado o projeto de implementação da fase de Vendas e Planejamento de Projeto do PV&DS *on-demand* instanciado para a Empresa 1. Alguns resultados qualitativos do projeto de mudança “implementação da

fase de Vendas” do Processo de V&DS *on-demand* foram: um aumento nas vendas de projetos de sistemas, a redução no tempo de criação de propostas para o cliente (em média, de 10 dias para 2 dias, segundo os vendedores da empresa) e melhor comunicação entre os colaboradores da área de venda e da área técnica da empresa.

A descrição detalhada da aplicação da Metodologia de Gestão de Mudança encontra-se na dissertação de mestrado de Costa (COSTA, 2006).

Com o lançamento da versão 1.2 do CMMI, foi realizado um novo mapeamento dos objetivos específicos definidos para cada área de processo do CMMI-DEV em relação às práticas-base da norma ISO/IEC 15504-5.

Durante a implantação de duas fases do modelo na Empresa 1 percebeu-se a necessidade de alterar a estrutura do modelo (seqüencial e linear) para retratar mais fielmente os projetos normalmente existentes.

Foram estudados vários modelos de processo de software existentes na literatura (MILLS et al. 1987; BOEHM et al., 1998; JACOBSON et al., 1999; PRESSMAN, 2006; SOMMERVILLE, 2007; ECLIPSE PROCESS FRAMEWORK COMMUNITY, 2008) para ser definida a nova estrutura do modelo de descrição de processo, agora denominado ProcSoftVD.

Selecionou-se o *framework* UP (*Unified Process*), pois propicia o desenvolvimento incremental e evolutivo do software, o que se acredita ser o mais voltado à realidade. Além disso, sua estrutura bi-dimensional permite a visualização das disciplinas ou *workflows* (neste trabalho de pesquisa denominado “áreas de conhecimento”) com maior ou menor ênfase em cada uma das fases que compõem o processo e permite, também, que possam ser selecionadas as áreas de conhecimento que a empresa acredita serem prioritárias em cada ciclo de melhoria. .

Para essa nova versão do Modelo ProcSoftVD foram definidos alguns requisitos, descritos na seção 5.2.1. Com o intuito de facilitar a visualização do relacionamento entre os elementos que compõem o modelo ProcSoftVD foi elaborado um meta-modelo, apresentado na seção 5.2.2. Na seção 5.2.3 são apresentadas as

duas perspectivas dessa nova versão do Modelo ProcSoftVD: fases e áreas de conhecimento.

5.2.1. Requisitos do ProcSoftVD

Os requisitos definidos nesta seção foram identificados durante a pesquisa realizada na Empresa 1. São eles:

- As atividades do processo devem ser agrupadas por fases e também por áreas de conhecimento para auxiliar na visualização, utilização e controle do processo;
- Cada atividade deve ter uma descrição geral e um detalhamento em forma de tarefas;
- Para cada atividade devem ser indicados os possíveis papéis desempenhados pelos responsáveis em executar a atividade;
- Para cada atividade devem ser indicados possíveis recursos (principalmente ferramentas computacionais) que poderiam auxiliar na realização da atividade;
- Para cada atividade devem ser indicados os artefatos de entrada e saída necessários;
- Para cada artefato do processo deve ser elaborado um *template* (formulário com a possível estrutura dos artefatos), a fim de facilitar para uma MPE saber o conteúdo que pode compor cada artefato;
- Deve-se permitir que o usuário tenha, além da visão textual das atividades do processo, uma visão gráfica. Essa visão gráfica permite a visualização dos relacionamentos das atividades (o que é difícil muitas vezes de ser identificado na visão textual);
- Deve-se permitir que o usuário tenha acesso às atividades de uma determinada fase do processo selecionada (tanto na visão gráfica quanto

na textual) e, a partir do acesso de uma delas, deve-se ter acesso às respectivas atividades e seus artefatos de entrada e saída. A partir do acesso de uma atividade deve ser possível visualizar as tarefas, o(s) recurso(s) necessário(s) para realização da atividade e o(s) papel(is) desempenhado(s) pelo(s) responsável(is);

- Deve-se permitir que o usuário tenha acesso às atividades, de acordo com uma determinada área de conhecimento selecionada;
- Deve-se permitir que o usuário tenha acesso aos objetivos específicos do CMMI e às práticas-base da norma ISO/IEC 15504-5 que são atendidos por cada uma das atividades pertencentes ao ProcSoftVD.

5.2.2. Meta-modelo do ProcSoftVD

O meta-modelo, apresentado na Figura 5.9, foi elaborado para facilitar a visualização do relacionamento entre os elementos que compõem o modelo ProcSoftVD.

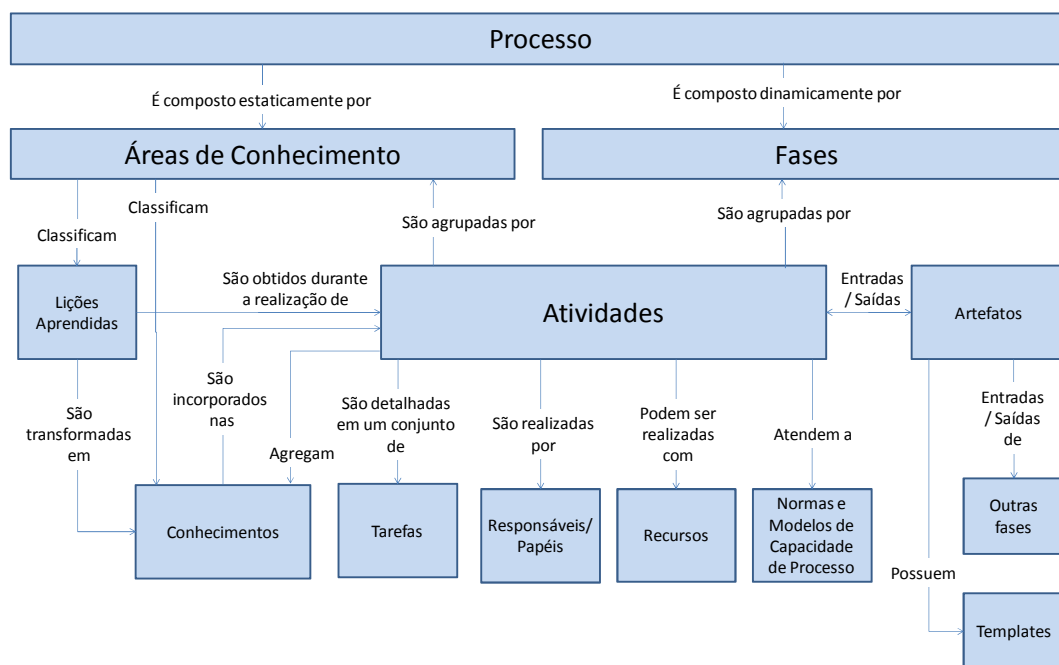


FIGURA 5.9 – META-MODELO DO PROCSoftVD

O ProcSoftVD é composto estaticamente por áreas de conhecimento e dinamicamente por fases. Assim, o modelo permite que o usuário tenha acesso às atividades do processo, agrupadas por fase ou agrupadas por áreas de conhecimento.

As atividades são detalhadas em um conjunto de tarefas, são realizadas por responsáveis que desempenham papéis na organização e podem ser realizadas com recursos (como ferramentas).

Para a realização das atividades têm-se como entrada artefatos e após a realização das atividades são criados/modificados artefatos (saídas). Esses artefatos possuem *templates* (formulários que apresentam uma possível estrutura para os artefatos; além disso, como suporte para os usuários, no próprio *template* há uma breve revisão bibliográfica relacionada ao artefato e em alguns casos, há alguns exemplos).

As atividades do processo atendem às normas e modelos de capacidade de processo (no caso, CMMI-DEV e ISO/IEC 15504-5). Durante a realização de cada uma das atividades, os responsáveis podem registrar lições aprendidas que são classificadas por áreas de conhecimento e podem ser transformadas em conhecimento integrante da memória organizacional da empresa que é utilizada como subsídio nos ciclos de melhoria do processo da empresa. Dessa forma, o conhecimento obtido é incorporado nas atividades do processo. Esses conhecimentos são classificados por áreas de conhecimento.

5.2.3. Perspectivas do ProcSoftVD

Como citado anteriormente, o ProcSoftVD possui duas perspectivas: fases e áreas de conhecimento.

a) Perspectiva - Fases

No modelo ProcSoftVD (Figura 5.10) foram adicionadas duas fases, além das fases sugeridas pelo UP: a fase de prospecção é sugerida devido o modelo abranger

não só o desenvolvimento, mas também a venda do software. E, a fase de negociação foi incluída no modelo, pois os modelos de processo existentes não abordam atividades relacionadas à negociação de um contrato com o cliente para o desenvolvimento do software. Essa fase de negociação é importante para se estabelecer o escopo do projeto e firmá-lo em um contrato.

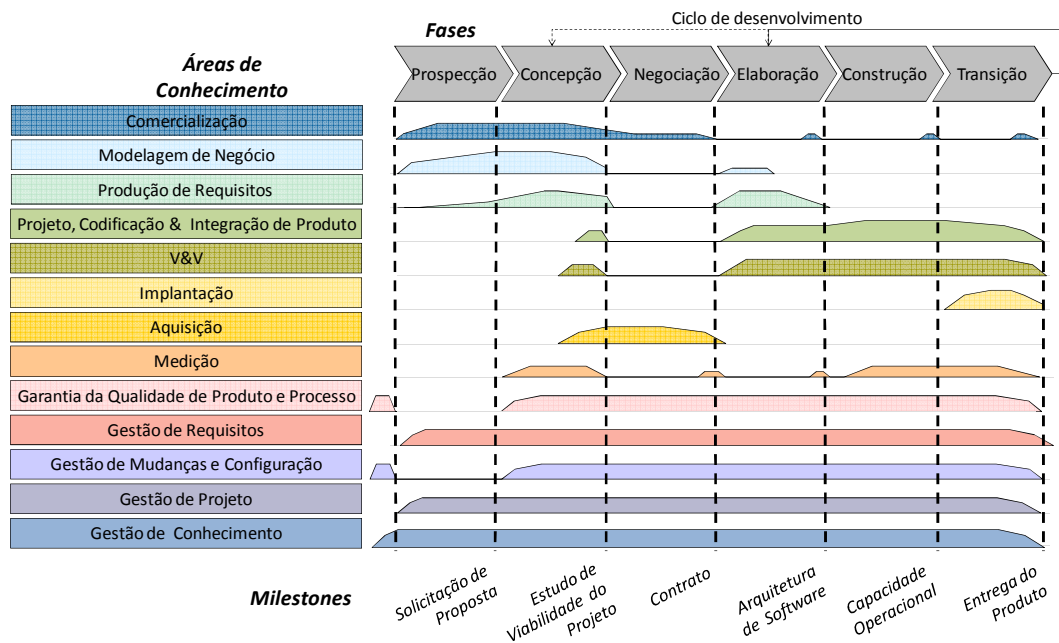


FIGURA 5.10 – MODELO PROCSoftVD

Uma vez realizada a busca de potenciais clientes e algum deles tenha solicitado uma proposta (fase de prospecção), inicia-se a fase de concepção. A fase de prospecção é realizada, normalmente, em apenas uma iteração.

A partir da fase de concepção até a fase de transição, forma-se o ciclo de desenvolvimento que ocorre a cada release (versão entregue ao cliente) do produto elaborado, ou seja, o software é elaborado de forma incremental e evolutiva. Entretanto, normalmente, o ciclo de desenvolvimento ocorre entre as fases de elaboração a transição, pois se estenderá para a fase de concepção nos casos onde o cliente solicita mudanças que implicam em criação de um novo contrato com novos acordos relacionados ao escopo do projeto, prazo de entrega e preço. Na fase de

concepção tem-se uma visão geral do produto e seu término é marcado pelo estudo de viabilidade do projeto. Nessa fase há tantas iterações quanto necessárias até que haja um acordo entre as partes envolvidas em relação ao contrato. Também, nessa fase é estimada a quantidade de ciclos de desenvolvimento necessária para o produto final ser concebido. Assim, para cada ciclo de desenvolvimento: na fase de elaboração os requisitos são detalhados e são planejados os testes; na fase de construção os requisitos são transformados em uma versão do produto e os testes planejados são executados; e na fase de transição é feito o teste de validação tanto pela empresa desenvolvedora, a partir dos critérios de validação estabelecidos pelo cliente, quanto pelo próprio cliente e é realizada a implantação da versão no cliente. São realizados os outros ciclos de desenvolvimento estimados para compor a versão final do produto. De acordo com os *feedbacks* (retornos) do cliente, as versões podem ter que passar por mais ciclos de desenvolvimento até estar de acordo com os requisitos estabelecidos pelo cliente.

Para cada fase realizada têm-se atividades com maior ênfase em determinadas áreas de conhecimento. Por exemplo, a área de conhecimento “Modelagem de Negócio” ocorre com maior ênfase nas fases de prospecção e concepção, mas se necessário podem ser executadas atividades dessa área na fase de elaboração, também. Em cada fase podem existir várias iterações (repetições) das atividades que a compõem, dependendo da necessidade.

A cada término de fase há um *milestone* (marco de referência) que indica o que se espera como produto final de cada fase, além de ser o marco onde são realizadas atividades relacionadas à garantia da qualidade do produto e do processo de software.

b) Perspectiva – Áreas de Conhecimento

As micro e pequenas empresas (MPEs) podem optar por iniciar a melhoria do processo de venda e desenvolvimento de software de sua empresa aos poucos, selecionando as áreas de conhecimento que acreditam ser de mais alta prioridade.

Para isso, utiliza-se o Método para Melhoria do Processo de Software voltado para MPEs (seção 5.3). Nesse caso, ela deve utilizar o modelo ProcSoftVD pela perspectiva estática, ou seja, pelas áreas de conhecimento. Assim, acessando uma área de conhecimento, serão apresentadas todas as atividades de cada uma das fases do modelo relacionadas de alguma maneira a essa área de conhecimento. Cada uma das atividades é identificada por uma sigla seguida de um número seqüencial que é reiniciado em cada fase (por exemplo: P01. Buscar contatos, CP02. Definir escopo do projeto). Na fase de prospecção, as atividades são identificadas pela sigla “P”, na fase de concepção por “CP”, na fase de negociação por “N”, na fase de elaboração por “E”, na fase de construção por “CT” e na fase de transição por “T”.

As áreas de conhecimento “GCf - Gestão de Mudanças e Configuração” e “GCo - Gestão de Conhecimento” permeiam todas as fases do modelo ProcSoftVD. Por isso, podem ser acessadas a qualquer momento em meio à realização das atividades.

Quanto às áreas de conhecimento consideradas nessa versão do ProcSoftVD, tem-se todas as áreas de processo do nível de maturidade 2 do CMMI-DEV e o restante das áreas de processo da categoria “Engenharia” do CMMI no nível de capacidade 2. Em alguns casos, o nome da área de conhecimento não é o mesmo do nome da área de processo do CMMI, pois o nome sugerido no modelo ProcSoftVD foi definido neste trabalho a partir da análise da descrição das áreas de processo do CMMI, dos processos utilizados na ISO/IEC 15504-5 e dos *workflows* definidos pelo *Unified Process* (UP). O Quadro 5.11 apresenta as áreas de conhecimento cobertas pelo ProcSoftVD e a associação com as áreas de processo do CMMI, com os processos da ISO/IEC 15504-5 e com os *workflows* do UP.

QUADRO 5.11– ASSOCIAÇÃO DAS ÁREAS DE CONHECIMENTO DO PROCSoftVD COM OUTROS MODELOS

Áreas de conhecimento do ProcSoftVD	CMMI-DEV	ISO/IEC 15504-5	Unified Process
Comercialização		- Prospecção do Fornecedor - Acordado Contratual	
Modelagem de Negócios			Modelagem de Negócios
Produção de Requisitos	- Desenvolvimento de Requisitos	- Elicitação de Requisitos - Análise de Requisitos do Sistema - Análise de Requisitos do Software	
Projeto, Codificação & Integração de Produto	- Solução Técnica - Integração de Produto	- Projeto do Software - Integração do Software - Integração do Sistema	- Implementação
V&V	- Verificação - Validação	- Verificação - Validação - Teste de Software - Teste de Sistema	- Teste
Implantação		- Entrega de Produto	- Implantação
Aquisição	- Gestão de Acordo com o Fornecedor	- Aquisição	
Medição	- Análise e Medição	- Medição	
Garantia da Qualidade de Produto e Processo	- Garantia da Qualidade de Produto e Processo	- Garantia da Qualidade de Produto e Processo	
Gestão de Requisitos	- Gestão de Requisitos		
Gestão de Mudanças e Configuração	- Gestão de Configuração	- Gestão de Configuração - Gestão de Solicitação de Mudanças	- Gerenciamento de Configuração e Mudança
Gestão de Projeto	- Planejamento de Projeto - Monitoramento e Controle de Projeto	- Gestão de Projeto	- Gerenciamento de Projetos
Gestão de Conhecimento		Gestão de Conhecimento	

Foi elaborado um questionário (Apêndice 1) aplicado a 5 (cinco) MPE's que desenvolvem software *on-demand* e 9 (nove) CPDs (Centro de Processamento de Dados) de empresas diversas os quais desenvolvem software *on-demand* para uso interno, que confirmou a importância de algumas áreas de conhecimento definidas no Modelo ProcSoftVD (Apêndice 2) e indicou a importância de outras áreas de conhecimento que serão melhor investigadas em um trabalho futuro.

Durante a análise desse questionário respondido pelas MPE's, notou-se que quase todas elas acreditam ser muito importante entender e modelar o negócio do cliente, entretanto, a maioria não o faz. Sendo assim, foi considerada também no ProcSoftVD a área de conhecimento "Modelagem de Negócio", indicada pelo modelo *Unified Process*. A área de conhecimento "Comercialização", apesar de não ser citada pelo CMMI-DEV, é bastante importante para as MPE's buscarem novos clientes, elaborarem/submeterem propostas ao cliente, negociarem um contrato claro e sem ambigüidades que especifique as expectativas, responsabilidades, entregas e compromissos de ambas as partes cliente e fornecedor, realizar a viabilidade financeira do projeto e controlar os pagamentos dos produtos/serviços vendidos pela empresa.

As áreas de conhecimento abordadas no ProcSoftVD e uma breve descrição de cada uma delas é feita, a seguir.

Comercialização: essa área de conhecimento foi criada para abordar (1) atividades relacionadas à prospecção de potenciais clientes para a empresa (o que não é coberto pelo CMMI, nem pela ISO/IEC 15504-5); (2) atividades relacionadas à elaboração/submissão de propostas ao cliente e negociação e aprovação de contrato sem ambigüidades que especifique as expectativas, responsabilidades, entregas e compromissos de ambas as partes cliente e fornecedor; (3) atividades relacionadas à viabilidade financeira do projeto e relacionadas ao pagamento dos produtos/serviços vendidos pela empresa. As atividades do item (2) foram definidas com subsídio dos processos “Prospecção do Fornecedor” e “Acordado Contratual” da ISO/IEC 15504-5.

Modelagem de Negócio: o objetivo dessa área de conhecimento é documentar os processos de negócio usando casos de uso de negócio²³, a fim de assegurar um entendimento comum entre todos os *stakeholders* (envolvidos) sobre as necessidades existentes no processo de negócio da organização cliente. Essa área de conhecimento teve sua origem no *Unified Process*.

Produção de Requisitos: tem o objetivo de produzir e analisar os requisitos do cliente, do produto e dos componentes de produto. Essa área de conhecimento teve sua origem relacionada à área de processo “Desenvolvimento de Requisitos” do CMMI-DEV e aos processos “Elicitação de Requisitos”, “Análise de Requisitos do Sistema” e “Análise de Requisitos do Software” da ISO/IEC 15504-5.

Projeto, Codificação & Integração de Produto: Engloba os processos “Projeto do Software”, “Integração do Software” e “Integração do Sistema” da 15504-5, o workflow “Implementação” do UP e as áreas de processo “Solução Técnica” e “Integração de Produto” do CMMI-DEV. Segundo a 15504-5, o processo “Projeto de

²³ Ver RUP – Rational Unified Process. Modelo integrado à ferramenta CASE Rational Rose.

Software” tem o objetivo de fornecer um *design* para o software que é implementado e pode ser verificado em confronto aos requisitos; o processo “Integração de Software” tem o objetivo de combinar as unidades de software, produzindo itens de software integrados, consistentes com o projeto de software, os quais demonstram que os requisitos funcionais e não-funcionais foram satisfeitos; e o processo “Integração do Sistema” tem como objetivo integrar os elementos do sistema (incluindo os itens de software, itens de hardware, operações manuais e outros sistemas, se necessário) para produzir um sistema completo que satisfaça ao projeto do sistema e às expectativas do cliente, expressadas em requisitos do sistema. Segundo o modelo CMMI-DEV, a área de processo “Solução Técnica” tem como objetivo projetar, desenvolver e implementar soluções para os requisitos, soluções essas que envolvem produtos, componentes de produtos e processos de ciclo de vida relacionados ao produto; e a área de processo “Integração do Produto” tem o objetivo de montar o produto, a partir dos componentes de produto, assegurar que o produto ao ser integrado funcione adequadamente, e entregar o produto.

V&V: Engloba as áreas de processo (CMMI) e processos (15504-5) “Validação” e “Verificação” e os processos “Teste de Software” e “Teste de Sistema” da 15504-5. O objetivo da área de processo/processo “Validação” é demonstrar que o produto ou componente do produto atenda ao uso pretendido quando colocado no ambiente destinado. Já o objetivo da área de processo/processo “Verificação” é assegurar que produtos de trabalho (e serviços, no caso do CMMI) selecionados alcancem seus requisitos especificados. Testes desempenham um papel extremamente importante em V&V (Verificação e Validação). Segundo o modelo 15504-5, o processo “Teste” pode ser realizado tanto no software quanto no sistema. O processo “Teste de Software” tem como objetivo confirmar que o produto de software integrado atende aos requisitos definidos. E, o processo “Teste de Sistema” tem o objetivo de assegurar que a implementação de cada requisito de sistema seja testada quanto à sua conformidade e que o sistema esteja pronto para entrega.

Implantação: o objetivo é entregar o produto produzido para os usuários finais, por meio das atividades de codificação, teste, empacotamento e instalação do software. Essa área de conhecimento teve sua origem no UP e também está relacionada ao processo “Entrega de Produto” da 15504-5.

Aquisição: essa área de conhecimento tem o objetivo de gerenciar a aquisição de produtos de fornecedores, sejam equipamentos ou até mesmo componentes de software do produto (no caso de terceirização do serviço). Exemplos de produtos e componentes de produtos que podem ser adquiridos pelo projeto: subsistemas (por exemplo, um sistema navegacional de uma aeronave), software, hardware, documentação (como manuais de instalação, de operação e do usuário). A origem dessa área de conhecimento está relacionada ao grupo de processo “Aquisição” da 15504-5 à área de processo denominada “Gestão de Acordo com Fornecedor” do CMMI-DEV.

Medição: essa área de conhecimento tem o objetivo de coletar e analisar dados relacionados aos produtos desenvolvidos e aos processos implementados dentro da organização por meio de projetos, a fim de dar um suporte efetivo à gestão dos processos e demonstrar objetivamente a qualidade dos produtos. Sua origem está relacionada à área de processo “Análise e Medição” do CMMI-DEV e do processo “Medição” da 15504-5.

Garantia da Qualidade de Produto e Processo: o objetivo dessa área de conhecimento é fornecer a garantia da qualidade dos processos e produtos de trabalho. Sua origem está relacionada tanto ao CMMI quanto à ISO/IEC 15504.

Gestão de Requisitos: o objetivo dessa área de conhecimento, que teve sua origem no CMMI-DEV, é gerenciar os requisitos dos produtos e componentes de produtos dos projetos e identificar inconsistências entre esses requisitos e os planos e produtos de trabalho dos projetos. Para isso, essa área trata do rastreamento dos requisitos em meio ao projeto e das mudanças desses requisitos. As mudanças

relacionadas aos requisitos utilizam algumas das atividades definidas pela área de conhecimento “Gestão de Mudanças e Configuração”.

Gestão de Mudanças e Configuração: essa área de conhecimento engloba a área de processo “Gestão de Configuração” do CMMI-DEV e os processos “Gestão de Configuração” e “Gestão de Solicitação de Mudanças” da 15504-5. O objetivo dessa área de conhecimento é estabelecer e manter a integridade de produtos de trabalho usando a identificação de configuração, controle de configuração, prestação de contas (explicação) do *status* da configuração e auditoria da configuração, além de assegurar que solicitações de mudanças no projeto sejam gerenciadas, rastreadas e controladas.

Gestão de Projetos: o objetivo dessa área de processo é identificar, estabelecer, coordenar e monitorar as atividades, tarefas e recursos necessários para um projeto produzir um produto e/ou serviço, no contexto dos requisitos e restrições de projetos. Engloba tanto o processo “Gestão de Projetos” da 15504-5 quanto as áreas de processo “Planejamento de Projeto” e “Monitoramento e Controle de Projeto” do CMMI.

Gestão de Conhecimento: tem como objetivo assegurar que o conhecimento, informação e habilidades individuais sejam coletadas, compartilhadas, reusadas e melhoradas por toda a organização. Sua origem está relacionada ao processo “Gestão de Conhecimento” da ISO/IEC 15504. A Figura 5.12 apresenta as atividades da área de conhecimento Gestão de Conhecimento, por meio da notação SADT adaptada, e o relacionamento dessas no contexto de melhoria de processo. Essa área de conhecimento permeia todo o processo de software, por isso, os responsáveis durante a realização de atividades podem ter alguma lição aprendida (conhecimento) a ser registrada. Essas lições aprendidas devem ser lapidadas por algum membro da empresa que desempenhe o papel de avaliador do conhecimento (mesmo que tenha subsídio da experiência de outros funcionários), integrando a memória organizacional da empresa. Uma vez que o conhecimento é lapidado, pode ser disseminado para todos os outros envolvidos no processo, por meio dos próximos ciclos de melhoria

definidos durante a aplicação do Método de Melhoria do Processo de Software (ProcSoftVD - Melhoria).

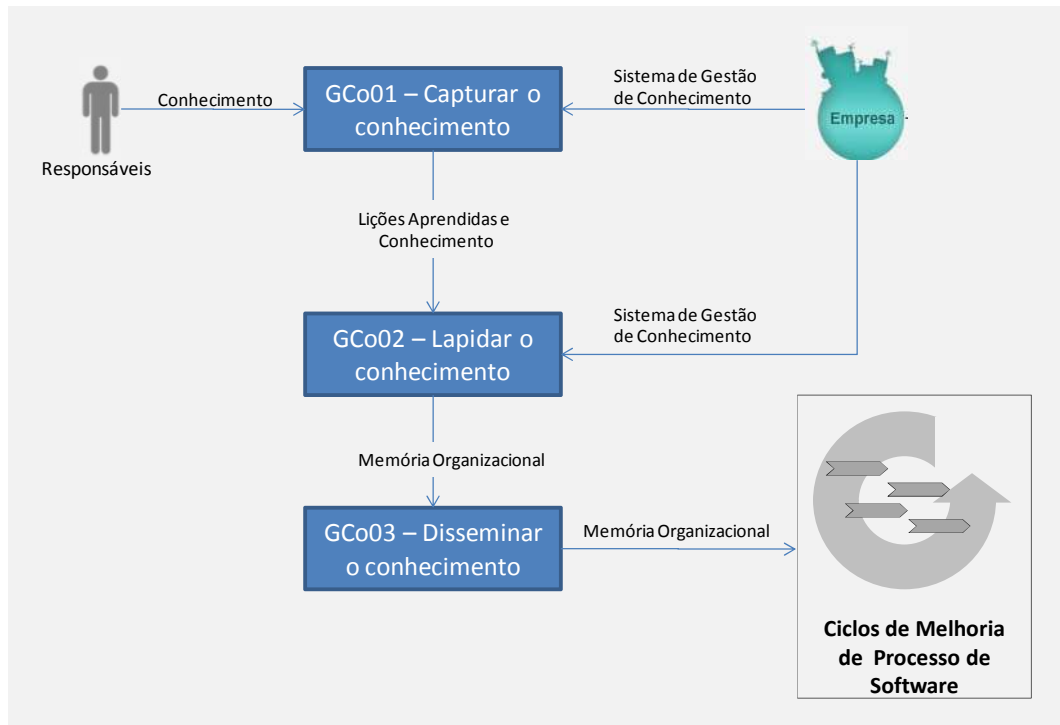


FIGURA 5.12– ATIVIDADES DA ÁREA DE GESTÃO DE CONHECIMENTO DO MODELO PROCSoftVD

O Modelo ProcSoftVD completo encontra-se nos Apêndices 3, 4 e 5 deste trabalho. O Apêndice 3 apresenta as atividades do modelo na visão textual, agrupadas por fases e por áreas de conhecimento, e na visão gráfica. Além disso, apresenta cada uma das atividades com sua respectiva descrição, responsáveis, artefatos de entrada e saída, recursos e tarefas. O Apêndice 4 apresenta todos os *templates* de artefatos criado/utilizado no modelo e o Apêndice 5 mostra o mapeamento das atividades do modelo ProcSoftVD com os objetivos específicos do CMMI-DEV e as práticas-base da ISO/IEC 15504-5.

5.3. Método de Melhoria do Processo de Venda e Desenvolvimento de Software *On-demand* para MPEs (ProcSoftVD – Melhoria)

Quando a empresa decide realizar algum tipo de melhoria em seus processos, é importante que essa melhoria seja analisada e planejada, antes de ser executada. Existem vários métodos de melhoria disponíveis na literatura (DEMING, 1986; MCFEELEY, 1996; INTERNATIONAL ORGANIZATION FOR STANDARDIZATION, 2003; WEBER et al., 2005b; SALVIANO, 2006; COSTA, 2006). Para utilizar o ProcSoftVD em um contexto de gestão de processos poderia ser utilizado, inicialmente, qualquer um desses métodos. Entretanto, realizando uma análise desses métodos (Quadro 5.13) percebeu-se uma complementaridade entre alguns deles e, por isso, foi elaborado o Método de Melhoria de Processo (ProcSoftVD – Melhoria), descrito nessa seção, a partir da combinação de três abordagens: ASPE/MSD (WEBER et al., 2005b), PRO2PI-WORK (SALVIANO, 2006) e Metodologia de Gestão de Mudanças (COSTA, 2006).

QUADRO 5.13 – QUADRO COMPARATIVO DOS MÉTODOS DE MELHORIA DE PROCESSO

ProcSoftVD - Melhoria	ASPE/MSC	PRO2PI-WORK	Metodologia de Gestão de Mudanças
Fase 1 - Diagnóstico do Processo de Software Atual e Definição do Perfil-Alvo	Diagnóstico do Processo de Software Atual	Definição do Perfil-Alvo	
1.1 Descrever o processo de software atual	X	X	
1.2 Identificar com os diretores da empresa os objetivos estratégicos do negócio, considerando metas de melhoria	X		
1.3 Definir os perfis-alvo do processo		X	
Fase 2 - Análise Estratégica e Planejamento da Mudança	Análise Estratégica		Planejamento da Mudança
2.1 Identificar, analisar e priorizar as áreas de conhecimento a serem melhoradas, com base nos perfis-alvo definidos na fase 1	X		
2.2 Definir um plano de ação	X (superficialmente)		X (detalhadamente)
Fase 3 - Definição das atividades da(s) área(s) de conhecimento	X		
3.1 Identificar as principais atividades que serão executadas durante a realização da área de conhecimento	X		
3.2 Identificar os papéis e as responsabilidades relacionadas a cada atividade selecionada	X		
3.3 Identificar os artefatos que serão consumidos ou produzidos durante a execução de cada atividade, e, caso existam, seus respectivos templates	X		
3.4 Identificar as medidas que devem ser coletadas durante a execução de uma atividade	X		
3.5 Identificar outras informações importantes, como critérios de entrada e de saída, métodos e ferramentas	X		
Fase 4 - Implantação da(s) área(s) de conhecimento	X		
4.1 Planejar como será avaliada a área de conhecimento durante a sua execução	X		
4.2 Definir escopo de utilização da área de conhecimento	X		
4.3 Treinar e motivar os participantes envolvidos	X		
4.4 Iniciar o uso do guia	X		
4.5 Analisar e interpretar os dados coletados durante a utilização do guia	X		
4.6 Apresentar os resultados da análise para a diretoria	X		

A primeira refere-se a uma abordagem para estabelecimento de processo de software em MPEs, a segunda refere-se a um método para estabelecimento do perfil de capacidade de processo para melhoria de processo em MPEs e a terceira refere-se a uma metodologia de gestão de mudanças que incorpora melhores práticas de gestão de projeto ao tratar as mudanças a serem realizadas por meio de projetos de mudanças.

O ProcSoftVD - Melhoria é composto por quatro fases, descritas a seguir, sugeridas pela ASPE/MSC (WEBER et al., 2005b) e adaptadas neste trabalho de pesquisa, a atividade de definição do perfil de processo é realizada utilizando a estratégia definida no PRO2PI-WORK (SALVIANO, 2006) e a atividade relacionada à definição de um plano de ação é realizada com base nas melhores práticas de gestão de projetos sugeridas por Costa (2006) em sua Metodologia de Gestão de Mudanças.

Fase 1. Diagnóstico do Processo de Software Atual e Definição do Perfil-Alvo

Nessa fase deve ser realizado o diagnóstico do processo de software atualmente utilizado na MPE. As atividades pertencentes a essa fase são:

- 1.1 Descrever o processo de software atual da empresa em alto nível. Para isso pode ser usada qualquer representação de modelagem de processo, tais como: SADT (*Structured Analysis and Design Technique*) (MCGOWAN & MARCA, 1987), IDEF0 (Integration definition for function modeling) (AIR FORCE, 1980), EPC (Event-driven process chain) (SCHEER, 1998), UML (Unified Modeling Language) e BPMN (*Business Process Modeling Notation*) (BUSINESS PROCESS MANAGEMENT INITIATIVE, 2008) ou até mesmo uma descrição textual.
- 1.2 Identificar com os diretores da empresa os objetivos estratégicos de negócio, considerando metas de melhoria.
- 1.3 Definir os perfis-alvo do processo, ou seja, quais são as áreas de conhecimento mais importantes para a empresa e qual o respectivo nível de capacidade que elas devem possuir para que os objetivos estratégicos e de melhoria sejam alcançados, considerando o escopo do modelo de referência utilizado (no caso, sugere-se o Modelo ProcSoftVD proposto neste trabalho de pesquisa). Isso pode ser realizado em uma reunião com pessoas representantes do nível gerencial e operacional da empresa. Para cada área de conhecimento devem ser realizadas as tarefas, a seguir, da forma mais rápida possível²⁴ :
 - apresenta-se a definição da área de conhecimento, conforme o modelo de referência utilizado; os principais sintomas típicos quando ela não é bem executada e os principais benefícios que justificam porque ela é importante.
 - discute-se e identifica-se a correspondência da área de conhecimento na organização.
 - identifica-se como a área de conhecimento é executada atualmente, ou seja, em qual nível de capacidade e justificativas.

²⁴ Em Salviano (2006) é sugerido não exceder 20 minutos.

- define-se a importância da área de conhecimento para os objetivos estratégicos da organização, observando os benefícios típicos de sua execução, com pontuação baixa, média ou alta, descrevendo as razões da pontuação.
- define-se o risco em continuar executando a área de conhecimento da forma como é executada atualmente na empresa, definindo uma pontuação (baixa, média ou alta), e descreve-se as razões da pontuação.
- posiciona-se cada área de conhecimento no quadro de importância versus risco e atribui-se uma pontuação para as combinações, com preferência para a importância. A pontuação, apresentada na Tabela 5.14, é sugerida em Salviano (2006).

TABELA 5.14 – PONTUAÇÃO PARA AS ÁREAS DE CONHECIMENTO (SALVIANO, 2006)

Risco	Baixo	Médio	Alto
Importância			
Alta	6	9	10
Média	4	7	8
Baixa	2	3	5

Fase 2. Análise Estratégica e Planejamento da Mudança

Esta fase tem por objetivo a definição e priorização das ações para o estabelecimento de áreas de conhecimento na empresa, baseado no diagnóstico e de acordo com os objetivos estratégicos e as metas de melhoria. Essas ações serão tratadas como um projeto de mudança. As atividades a serem realizadas nessa fase são:

2.1 Identificar, analisar e priorizar as áreas de conhecimento a serem melhoradas, com base nos perfis-alvo definidos na fase 1. As áreas de conhecimento a serem priorizadas são as com maior pontuação, atribuída durante a atividade 1.3. Dentre essas áreas de conhecimento selecionadas pode(m) ser definida(s), ainda, a(s)

mais prioritária(s) a ser(em) considerada(s) nos primeiros ciclos de melhoria. Para isso devem ser considerados benefícios, custos estimados, interdependência entre áreas de conhecimento, metas de negócio e melhoria, frequência de uso da área de conhecimento, grau de divergência entre seus executores, número de atores envolvidos, etc. Poucas áreas de conhecimento devem ser selecionadas para serem estabelecidas em cada ciclo de melhoria.

2.2 Definir um plano de ação. Para isso, deve-se:

- Definir um patrocinador dentro da organização cujas responsabilidades englobam assegurar o comprometimento da alta gerência e assegurar verba para que os recursos necessários durante o processo de mudança sejam fornecidos.
- Definir o escopo do projeto de mudança: deve-se definir, em detalhes, as entregas do projeto de mudança e as ações necessárias para criar essas entregas. Devem ser documentados características e limites do projeto, além dos métodos de aceitação e controle do escopo. O escopo do projeto de mudança deve fornecer um entendimento comum a todos os *stakeholders* e descrever os principais objetivos do projeto.
- Criar a WBS (*Work Breakdown Structure*): o objetivo desta atividade é dividir as principais entregas do projeto e os pacotes de trabalho em componentes menores, ou seja, em atividades que se deseja controlar e monitorar durante o projeto.
- Elaborar o cronograma do projeto: deve-se estabelecer a programação do projeto, na qual estarão definidas as datas de início e término de cada atividade planejada, a precedência entre elas, as pessoas responsáveis por cada uma delas e os recursos necessários para a sua execução. Quando o cronograma estiver elaborado, deve-se nivelar o esforço das pessoas da equipe em relação à realização das atividades para evitar sobrecarga de algum colaborador, ou criar gargalos no desenvolvimento do projeto de

mudança, decorrentes de excesso de atividades atribuídas à mesma pessoa.

- Avaliar os riscos: é importante identificar os principais riscos do projeto de mudança e administrar os riscos. Uma vez identificados, para administrá-los é preciso definir a probabilidade de ocorrência e o impacto de cada um deles. Para os riscos de maior importância e probabilidade de ocorrência, deve ser criado um plano de ação para sua mitigação e um plano de contingência.
- Garantir a infra-estrutura: deve-se garantir “como” a empresa vai dar suporte à mudança. Devem ser definidos os treinamentos para as equipes, como elas serão avaliadas, quais as ferramentas necessárias para a execução da mudança, e se a infra-estrutura existente é adequada.

Fase 3. Definição das atividades da(s) área(s) de conhecimento

Nessa fase deve ser realizada a definição/adaptação das atividades das áreas de conhecimento, selecionadas no projeto de mudança anteriormente planejado, de forma explícita e descritiva, para que as pessoas se orientem durante a execução dessas atividades. Para isso, poderiam ser utilizados normas e modelos de referência, tais como ISO/IEC 12207, CMMI, ISO/IEC 15504 e, até mesmo, processos proprietários como o RUP. Neste trabalho de pesquisa, sugere-se a utilização do Modelo ProcSoftVD.

As atividades que compõem essa fase são:

- 3.1 Identificar as principais atividades que serão executadas durante a realização da área de conhecimento. A seqüência de execução que elas acontecem e os desvios condicionais existentes podem ser visualizados por meio de um fluxograma ou de diagramas SADT.
- 3.2 Identificar os papéis e as responsabilidades relacionados a cada atividade selecionada.

- 3.3 Identificar os artefatos que serão consumidos ou produzidos durante a execução de cada atividade, e, caso existam, seus respectivos *templates*.
- 3.4 Identificar as medidas que devem ser coletadas durante a execução de uma atividade, pois estabelecem quais dados quantitativos e qualitativos serão coletados para dar suporte à gerência de projetos, a melhoria e a garantia da qualidade.
- 3.5 Identificar outras informações importantes, como critérios de entrada e de saída, métodos e ferramentas.

Fase 4. Implantação da(s) área(s) de conhecimento

Nessa fase a área de conhecimento definida anteriormente deve ser institucionalizada e avaliada. Para isso, deve-se garantir que todos os envolvidos conheçam e utilizem a área de conhecimento e sejam coletados dados que dêem informações sobre os resultados obtidos. As atividades a serem realizadas são:

- 4.1 Planejar como será avaliada a área de conhecimento durante a sua execução. São definidas medidas que devem ser coletadas durante o uso da área de conhecimento, visando à análise das metas que foram definidas na fase de análise estratégica. Para isso, pode ser usado o método GQM (*Goal-Question-Metric*) (BASILI & WEISS, 1984).
- 4.2 Definir o escopo de utilização da área de conhecimento, ou seja, decidir se o guia elaborado será utilizado inicialmente em todos os projetos da empresa ou apenas será executado um projeto piloto para avaliação.
- 4.3 Treinar e motivar os participantes envolvidos, dentro do escopo que foi definido, para que estejam aptos a desempenhar suas atividades. O objetivo desta atividade é mobilizar as pessoas integrantes da equipe do projeto de mudança, preparar e aplicar os treinamentos definidos como sendo requisitos para garantir a eficácia do projeto de mudança.

4.4 Iniciar o uso do guia. Durante a utilização, as dúvidas que surgem são esclarecidas, sugestões e observações são anotadas e as medidas definidas durante o planejamento da avaliação são coletadas.

4.5 Analisar e interpretar os dados coletados durante a utilização do guia. Essa atividade deve ser realizada pelo engenheiro de processo junto a alguma outra pessoa da organização.

4.6 Apresentar os resultados da análise para a diretoria, para a tomada de decisão relacionada à iniciação de um novo ciclo de melhoria – projeto de mudança.

5.4. Publicação web do Modelo de Gestão do Processo de Venda e Desenvolvimento *On-demand* para MPEs

O ProcSoftVD – Gestão encontra-se disponível no endereço www.numa.org.br/GProcSoftVD. Na página principal (Figura 5.15) encontra-se uma breve introdução sobre o modelo. Há um menu do lado esquerdo com quatro *links*: “Visão Gráfica”, “Visão Textual”, “Mapeamento com CMMI e ISO/IEC 15504” e “ProcSoftVD - Melhoria”. O primeiro *link* “Visão Gráfica” refere-se à apresentação do modelo em forma gráfica, onde as atividades serão representadas com a notação SADT (MCGOWAN & MARCA, 1987) adaptada.

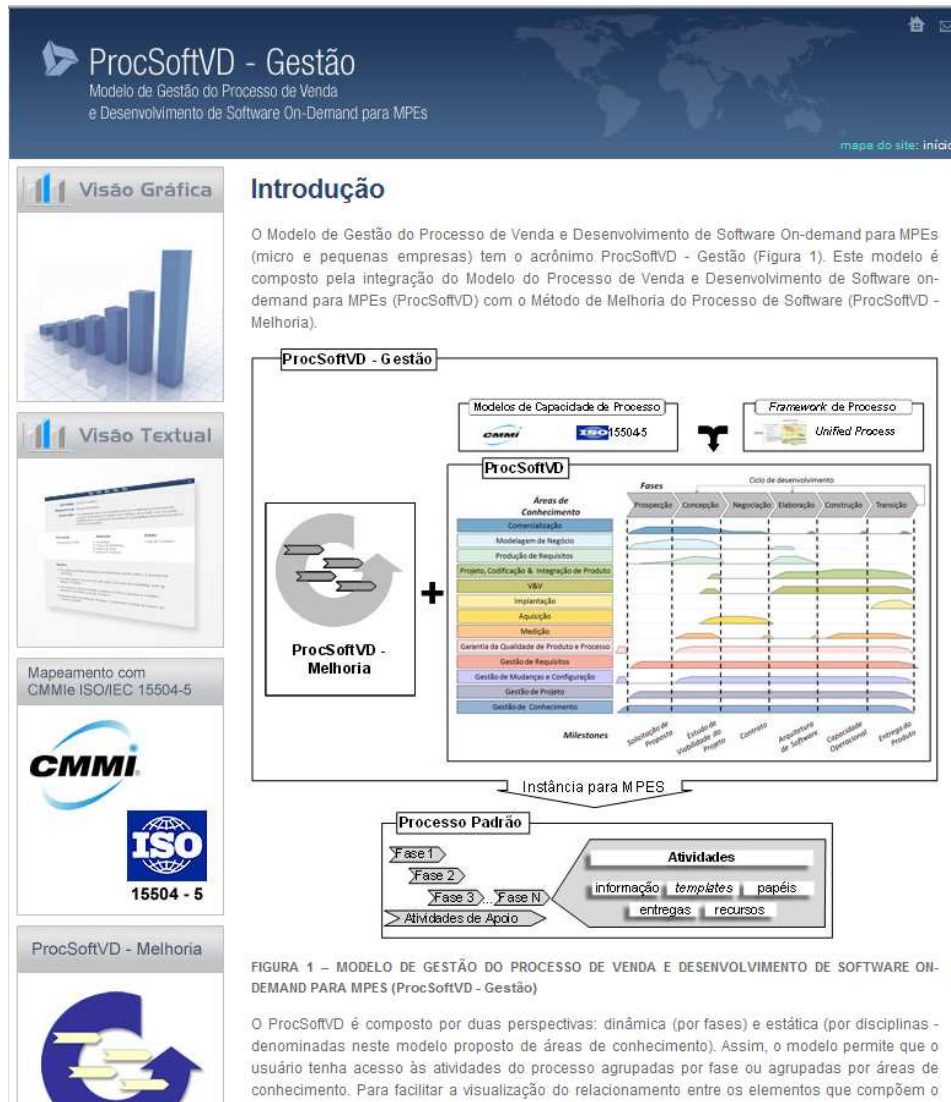


FIGURA 1 – MODELO DE GESTÃO DO PROCESSO DE VENDA E DESENVOLVIMENTO DE SOFTWARE ON-DEMAND PARA MPES (ProcSoftVD - Gestão)

FIGURA 5.15 – PÁGINA PRINCIPAL DO GPROCSoftVD

Acessando o link “Visão Gráfica” é possível visualizar as atividades do modelo e a seqüência entre elas. A Figura 5.16 apresenta um exemplo de visão gráfica.

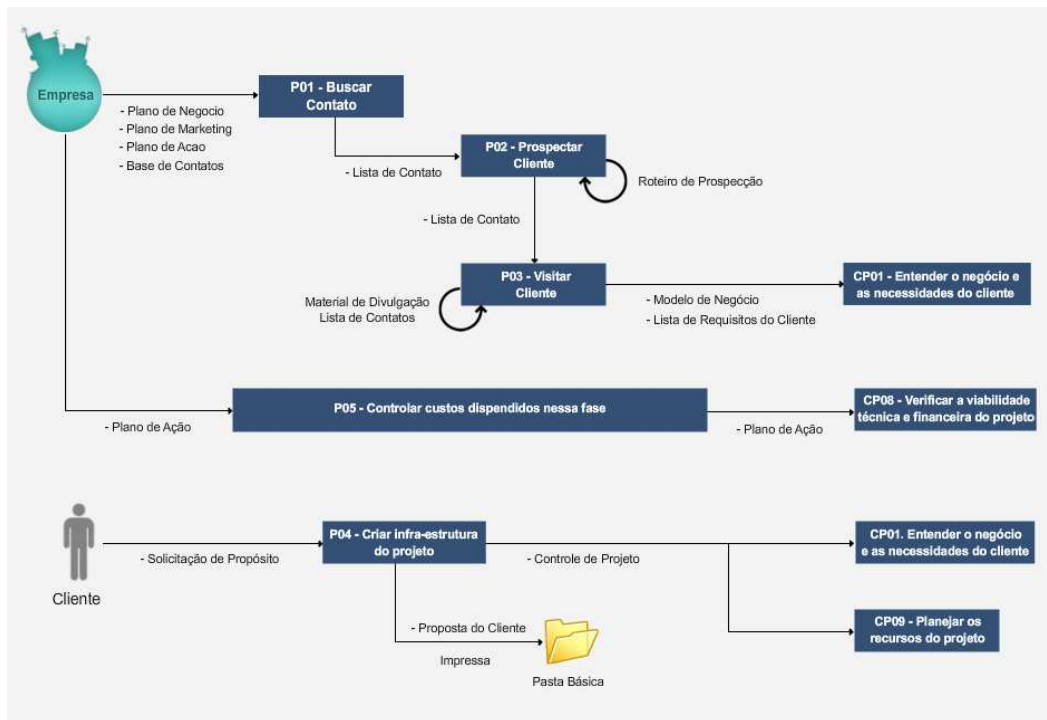
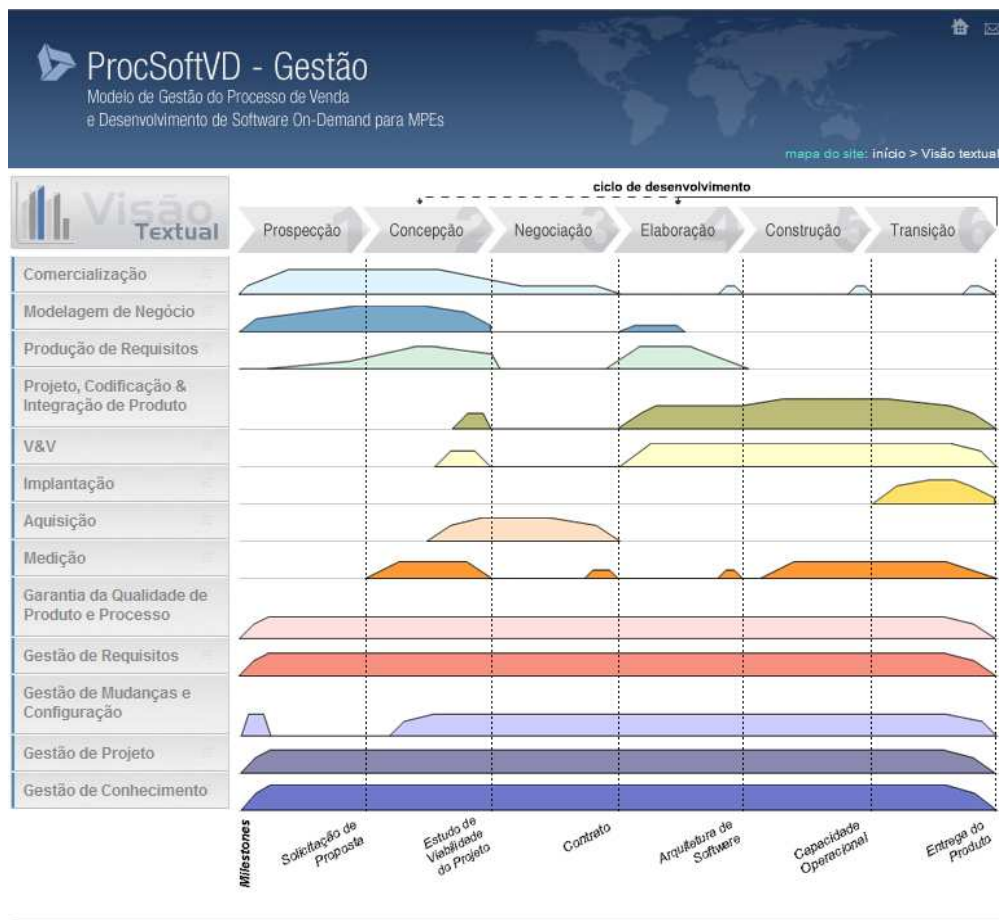


FIGURA 5.16 – EXEMPLO DE VISÃO GRÁFICA – FASE DE PROSPECÇÃO

Acessando o *link* “Visão Textual” é possível ter uma visão dinâmica (por fases) e estática (por áreas de conhecimento) do modelo (Figura 5.17). Clicando nas fases, tem-se acesso a uma página que mostra todas as atividades do modelo com suas respectivas entradas e saídas. Um exemplo é apresentado na Figura 5.18. Clicando nas atividades, tem-se acesso a uma página com todos os detalhes dessa atividade. Nessa página, as entradas e saídas que possuírem *templates* terão um *link* e os documentos com *templates* citados nas tarefas, serão representados com a marcação “(T)” (Figura 5.19). Os *templates* são formulários com uma possível estrutura para o artefato correspondente e, em alguns deles há exemplos e até mesmo uma breve revisão bibliográfica para explicar os conceitos pertinentes ao artefato.



ProcSoftVD - Gestão – Modelo de Gestão do Processo de Venda e Desenvolvimento de Software On-Demand para MPES

FIGURA 5.17 – VISÃO TEXTUAL: PERSPECTIVA POR FASES E POR ÁREAS DE CONHECIMENTO



ProcSoftVD - Gestão – Modelo de Gestão do Processo de Venda e Desenvolvimento de Software On-Demand para MPES

FIGURA 5.18 – EXEMPLO DE UMA ATIVIDADE - FASE DE PROSPECÇÃO

ProcSoftVD - Gestão
Modelo de Gestão do Processo de Venda e Desenvolvimento de Software On-Demand para MPES

mapa do site: início > prospecção > P01 - Buscar contatos

Visão Textual

ciclo de desenvolvimento

Prospecção → Concepção → Negociação → Elaboração → Construção → Transição

Comercialização
Modelagem de Negócio
Produção de Requisitos
Projeto, Codificação & Integração de Produto
V&V
Implantação
Aquisição
Medição
Garantia da Qualidade de Produto e Processo
Gestão de Requisitos
Gestão de Mudanças e Configuração
Gestão de Projeto
Gestão de Conhecimento

Atividade: P01 - Buscar contatos
Responsável: Equipe de Vendas
Descrição: Essa atividade tem como objetivo buscar na internet ou na base de contatos da empresa os potenciais clientes, de acordo com os perfis estabelecidos no plano de marketing e na estratégia da empresa para o período em questão.

Recursos:

- Ferramentas de relacionamento com cliente (CRM), tais como: VTIGER, DBMKT, newsletter, e-mail marketing, mecanismos de buscas, websites

Entradas:

- Plano de Negócios (já deve existir na empresa)
- Plano de Marketing (já deve existir na empresa)
- Plano de Ação (já deve existir na empresa)
- Base de contatos (já deve existir na empresa)

Saídas:

- Lista de Contatos

Tarefas:

- Executar as atividades para busca de potenciais clientes (item 3.2 do "Plano de Marketing (T)"):
 - Executar a etapa 1 definida no item "Plano de Ação (T)" (item 4 do "Plano de Marketing (T)", a fim de validar os contatos, com base no planejamento definido na planilha "Plano de Ação (T)" do arquivo "Plano de Ação (T)".
 - Criar planilha Lista de Contatos (conforme GCf01) e registrar os contatos validados na planilha "Lista de Contatos (T)", a partir da "Base de Contatos" da empresa.
 - Registrar a data de realização da etapa 1 descrita no item 4 do "Plano de Marketing (T)" na planilha "Controle de Visitas" do arquivo "Lista de Contatos (T)", para todos os contatos validados.

ProcSoftVD - Gestão – Modelo de Gestão do Processo de Venda e Desenvolvimento de Software On-Demand para MPES

FIGURA 5.19 – EXEMPLO DE DETALHAMENTO DE ATIVIDADE

Acessando o *link* “Visão Textual” também é possível visualizar as áreas de conhecimento do modelo no menu à esquerda. Clicando nas áreas de conhecimento, tem-se acesso a uma página que mostra todas as atividades do modelo que possuem algo a ser realizado relativo àquela área de conhecimento (Figura 5.20). Clicando nas atividades, você tem acesso a uma página com todos os detalhes dessa atividade, como mostrado na Figura 5.19. Essa opção de visualizar as atividades por área de conhecimento é interessante para as MPES que definiram um perfil de capacidade de processo a ser alcançado, ou seja, quais são as áreas de conhecimento de mais alta

prioridade a serem melhoradas em sua empresa, por meio da aplicação do “Modelo para Melhoria do Processo de Software voltado para MPEs” (acessado por meio da página inicial do site, ao clicar no *link* “Melhoria do Processo de Software voltado para MPEs” no menu do lado esquerdo).



FIGURA 5.20 – ATIVIDADES DA ÁREA DE CONHECIMENTO “GESTÃO DE CONHECIMENTO”

O mapeamento de todas as atividades do modelo ProcSoftVD com os objetivos específicos do CMMI-DEV e práticas-base da ISO/IEC 15504-5 está disponível por meio do acesso ao *link* “Mapeamento com CMMI e ISO/IEC 15504” no menu do lado esquerdo da página inicial do site (Figuras 5.21 e 5.22).



ProcSoftVD - Gestão

Modelo de Gestão do Processo de Venda e Desenvolvimento de Software On-Demand para MPes




[mapa do site: inicio > Mapeamento com CMMI e ISO/IEC 15504-5](#)

Visão Gráfica



Visão Textual



Mapeamento com CMMI e ISO/IEC 15504-5




ProcSoftVD - Melhoria



Mapeamento com CMMI e ISO/IEC 15504-5

Fases

- Prospecção
- Concepção
- Negociação
- Elaboração
- Construção
- Transição



LEGENDA

Áreas de Conhecimentos

- Aquisição
- Garantia da Qualidade de Produto e Processo
- Gestão de Mudanças e Configuração
- Gestão de Conhecimento

Prospecção [topo](#)

P03 - Visitar cliente	
CMMI	ISO/IEC 15504-5
<ul style="list-style-type: none"> RD - SP 1.1 Elicit Needs 	<ul style="list-style-type: none"> ENG.1.BP1: Obtain customer requirements and requests. Obtain and define customer requirements and requests through direct and continuous solicitation of customer and user input.

Concepção [topo](#)

Cp01. Entender o negócio e as necessidades do cliente	
CMMI	ISO/IEC 15504-5
<ul style="list-style-type: none"> RD - SP 1.1 Elicit Needs RD - SP 1.2 Develop the Customer Requirements 	<ul style="list-style-type: none"> ENG.1.BP1: Obtain customer requirements and requests. Obtain and define customer requirements and requests through direct and continuous solicitation of customer and user input.

FIGURA 5.21 – MAPEAMENTO DO PROCSoftVD COM CMMI E ISO/IEC 15504-5

Cp02 - Definir escopo do projeto	
CMMI	ISO/IEC 15504-5
<ul style="list-style-type: none"> ● PP - SP 1.1 Estimate the Scope of the Project ● PP - SP 1.3 Define Project Lifecycle (este objetivo específico é atendido ao ser utilizado esse modelo de processo proposto neste trabalho de pesquisa, como ciclo de vida do projeto) ● PP - SP 2.7 Establish the Project Plan ● RD - SP 1.1 Elicit Needs ● RD - SP 1.2 Develop the Customer Requirements 	<ul style="list-style-type: none"> ● MAN.3.BP1: Define the scope of work. Identify the project's objectives, motivation and boundaries and define the work to be undertaken by the project. ● MAN.3.BP2: Define project life cycle. Define a life cycle and strategy for the project, appropriate to its scope, context, magnitude and complexity. (esta prática base é atendida ao ser utilizado esse modelo de processo proposto neste trabalho de pesquisa, como ciclo de vida do projeto) ● MAN.3.BP10: Establish project plan. Define and maintain project master plan and other relevant plans to cover the project scope and goals, resources, infrastructure, interfaces and communication mechanisms. ● ENG.1.BP1: Obtain customer requirements and requests. Obtain and define customer requirements and requests through direct and continuous solicitation of customer and user input.

FIGURA 5.22 – EXEMPLO DE MAPEAMENTO DA ATIVIDADE “DEFINIR ESCOPO DO PROJETO”

Os Apêndices 3, 4 e 5 apresentam o modelo ProcSoftVD por completo.

5.5. Considerações Finais

Este capítulo apresentou o processo de desenvolvimento da principal entrega deste trabalho de pesquisa: o Modelo de Gestão do Processo de Venda e Desenvolvimento de Software *On-demand* para MPEs (ProcSoftVD – Gestão).

No próximo capítulo são apresentadas as aplicações do Modelo ProcSoftVD - Gestão e os resultados obtidos durante essas aplicações.

6. Aplicações e Resultados do Modelo ProcSoftVD - Gestão

6.1. Considerações Iniciais

O ProcSoftVD - Gestão foi aplicado em uma MPE (seção 6.2) e o ProcSoftVD foi verificado em dois tipos de avaliação: análise dos pontos fortes e fracos do modelo por parte de profissionais da área (seção 6.3) e análise do modelo quanto às características de qualidade de produto (seção 6.4).

6.2. Aplicação do ProcSoftVD - Gestão

A aplicação foi realizada em uma MPE (Empresa 2) que possui uma carteira de 40 (quarenta) clientes, aproximadamente, e possui 3 (três) funcionários.

Após a aplicação do método de melhoria de processo ProcSoftVD - Melhoria, descrita no Apêndice 6, as áreas de conhecimento Modelagem de Negócios e Produção de Requisitos foram selecionadas como mais prioritárias e, por isso, foram implantadas no primeiro ciclo de melhoria nessa MPE.

Foi relatado pelo desenvolvedor que aplicando as atividades e *templates* sugeridos para essas áreas de conhecimento, pôde perceber o quão mais fácil é desenvolver um software quando se entende realmente o negócio do cliente e quando esse negócio é documentado de alguma forma que lhe dê subsídio para tirar eventuais dúvidas durante a definição dos requisitos do software a ser desenvolvido.

Anteriormente à aplicação dessas áreas de conhecimento, não se tinha nenhuma sistemática em relação a elas na empresa, além de não se ter *templates* para facilitar o registro das informações obtidas com o cliente. O desenvolvedor da empresa também relatou que o tempo investido para realizar as atividades de modelagem de negócios e produção de requisitos foi compensado por meio da

minimização de erros encontrados no software ao realizar os testes de mesa, da forma como realizam na empresa.

6.3. Análise dos Pontos Fortes e Fracos do Modelo ProcSoftVD

Foi realizada uma análise do modelo ProcSoftVD por 6 (seis) profissionais da área de computação, a partir de um questionário (Apêndice 7).

Os critérios a serem atendidos para selecionar esses profissionais foram: ter experiência como analista de sistemas e desenvolvedor de programas e ter conhecimentos a respeito da área de engenharia de software. O perfil de cada um dos profissionais é descrito, a seguir.

- Um profissional (**P1**) com graduação na área de computação e mestrado na área de informática educacional que já trabalhou em empresas desempenhando as funções de analista de sistemas/programador e até mesmo de gerente de CPD. Atualmente, é docente em uma universidade e leciona disciplinas, tais como: Fundamentos de Sistemas de Informação, Gestão da Informação e de Sistemas de Informação, Inglês Técnico, Introdução à Ciência da Computação e Engenharia de Software.

- Um profissional (**P2**) com graduação e mestrado na área de computação que já trabalhou em um CPD de uma universidade desempenhando as funções de analista de sistemas/programador. Atualmente, é docente em uma universidade e leciona disciplinas, tais como: Algoritmos, Engenharia de Software, Banco de Dados, Linguagens de Programação (Delphi, JavaScript).

- Um profissional (**P3**) com graduação e mestrado na área de computação que atua como coordenador do Curso em Ciência da Computação em uma universidade e docente de disciplinas, tais como: Análise e Projeto de Sistemas Orientados a Objetos, Engenharia de Software, Banco de Dados, Algoritmos e Linguagens de Programação.

- Um profissional (**P4**) com graduação, mestrado e doutorado na área de computação que já trabalhou desempenhando o papel de analista de sistemas/programador em uma empresa. Atualmente, é docente em uma universidade pública e leciona as disciplinas: Engenharia de Software, Linguagens de Programação (JAVA), Compiladores e Teoria da Computação. Já atuou como programador em empresas.

- Um profissional (**P5**) que está finalizando seu curso de graduação em Sistemas de Informação, entretanto, é proprietário de uma empresa de desenvolvimento de software que está no mercado há 8 anos e tem uma carteira de, aproximadamente, 40 clientes de um determinado produto da empresa – a MPE onde foram aplicadas as áreas de conhecimento “Modelagem de Negócios” e “Produção de Requisitos”.

- Um profissional (**P6**) que está finalizando seu curso de graduação em Sistemas de Informação, tem uma empresa de desenvolvimento de software incubada na INTEPP (Incubadora Tecnológica de Presidente Prudente) e adotou o modelo de processo, elaborado nesse trabalho de pesquisa, como guia para orientar nas atividades de venda e desenvolvimento de software na sua empresa.

As respostas desses 6 (seis) profissionais da área de computação podem ser visualizadas no Apêndice 8. Os resultados consolidados, a partir dessas respostas, são:

- **Quanto à Área de Conhecimento Comercialização:** o modelo cobre atividades relacionadas à comercialização as quais a organização desenvolvedora deve se preocupar. Essas atividades fornecem condições para o gerenciamento do contato com o cliente, transmitindo eficiência e credibilidade perante o cliente, além de abranger aspectos formais da comercialização sem exageros, por meio de uma linguagem de fácil entendimento.

- **Quanto à Área de Conhecimento Modelagem de Negócios:** as tarefas que incluem a modelagem de negócio estão claramente definidas, com foco em aspectos fundamentais dessa área de conhecimento. Há clareza na exposição das atividades e riqueza de detalhes por meio dos *templates* que mostram exemplos e sugestões.
- **Quanto à Área de Conhecimento Produção de Requisitos:** “Tempo despendido em modelagem, análise e produção de requisitos é um excelente investimento, e não um gasto” - comentário realizado por um dos profissionais. O modelo apresenta as atividades e *templates* relacionados à produção de requisitos de modo completo, seguindo diretrizes do padrão de documentação de requisitos IEEE 830 e atendendo aos objetivos específicos do CMMI em relação à área de processo “Desenvolvimento de Requisitos”. Entretanto, um dos profissionais relatou que o material para ser lido, analisado, documentado e empreendido, para autônomos e equipes pequenas, é bastante extenso.
- **Quanto à Área de Conhecimento Projeto, Codificação & Integração de Produto:** um dos profissionais relatou que “um software deve ser bem projetado para ser desenvolvido de forma a permitir uma fácil manutenção e evolução dele, pois os custos de manutenção são muito superiores ao custo de desenvolvimento”. O modelo apresenta um vasto material de apoio para a transformação dos requisitos em uma solução de projeto, fornecendo subsídios para o início da codificação por parte dos responsáveis pelo desenvolvimento do software.
- **Quanto à Área de Conhecimento V&V:** as atividades de V&V, especialmente os testes, têm grande importância durante o desenvolvimento do software. Essas atividades foram bem distribuídas nas fases do modelo ProcSoftVD, tendo uma presença marcante no modelo.

- **Quanto à Área de Conhecimento Implantação:** o modelo apresenta atividades relacionadas à implantação dos *releases* do produto ao cliente de forma completa, facilitando o gerenciamento das entregas ao cliente.
- **Quanto à Área de Conhecimento Aquisição:** o modelo contribui com essa área de conhecimento, pois mostra com clareza o que pode ser realizado para realizar a cotação e análise dos melhores fornecedores para viabilizar o projeto de desenvolvimento para a empresa e para ter controle da qualidade dos produtos/serviços adquiridos.
- **Quanto à Área de Conhecimento Medição:** o modelo fornece subsídios para se obter dados relacionados aos projetos da empresa, dados como tempo estimado e concretizado, valor cobrado, entre outros, que serão utilizados para estimativas em projetos futuros. Além disso, deixa evidente a importância da definição de critérios de medição e coleta de medidas para o aprimoramento do processo e para a garantia da qualidade do produto.
- **Quanto à Área de Conhecimento Garantia da Qualidade de Produto e Processo:** a garantia da qualidade tanto do produto quanto do processo foi apontada no modelo em todas as etapas de desenvolvimento, o que representa uma potencialidade do modelo.
- **Quanto à Área de Conhecimento Gestão de Requisitos:** a gestão de requisitos é abordada no modelo de maneira bem articulada e completa. Durante o avanço nas fases do modelo, nas atividades onde podem ser realizadas modificações nos requisitos, o modelo apresenta atividades de rastreamento dos requisitos e direciona as modificações para serem tratadas pelas atividades de gestão de mudanças e configuração. Um ponto vulnerável detectado foi o caso de não estar explícito no modelo o gerenciamento dos requisitos relacionados aos possíveis componentes (de software, por exemplo) adquiridos de fornecedores.

- **Quanto à Área de Conhecimento Gestão de Projetos:** é importante entender o escopo do projeto e fazer o planejamento preliminar desse escopo, a fim de se ter uma visão macro em relação ao que se pretende atingir com o produto resultante do projeto, quanto custará, quanto tempo demorará para ser produzido e qual o retorno financeiro desse projeto. E, também, é importante fazer o monitoramento e controle desse projeto para que se possa tomar medidas reativas e até mesmo pró-ativas. O modelo ProcSoftVD atende a esses requisitos.
- **Quanto à Área de Conhecimento Gestão de Mudanças e Configuração:** o modelo mostra os cuidados que devem ser tomados e os controles necessários ao ser realizada alguma modificação no projeto em desenvolvimento. Entretanto, seria interessante que o modelo explicitasse alguma forma de *help-desk* para auxiliar no controle das solicitações de modificações, principalmente do cliente.
- **Quanto à Área de Conhecimento Gestão de Conhecimento:** o modelo mostra uma forma de gerenciar o conhecimento de todos os colaboradores da empresa, fazendo com que o conhecimento e experiência de um colaborador sejam disseminados para outros interessados.

Além de identificar pontos fracos e fortes do modelo em relação a cada área de conhecimento, os seis profissionais responderam a quatro questões. As respostas para essas questões encontram-se no Apêndice 9. Para os profissionais que não conheciam o CMMI e a ISO/IEC 15504, foi realizada uma apresentação, expondo os conceitos, estrutura e funcionamento de cada um deles.

Quanto à análise do mapeamento das atividades do modelo ProcSoftVD para com os objetivos específicos do CMMI e as práticas-base da ISO/IEC 15504-5, foram realizadas entrevistas desestruturadas com os profissionais (P1, P2, P3, P4, P5 e P6). De forma geral, esses profissionais acreditam que esse mapeamento possa ser

bastante interessante para o caso de empresas que desejam futuramente iniciar um processo de certificação pelo CMMI ou MPS.BR, e até mesmo para que as empresas tenham consciência que as atividades sugeridas pelo modelo ProcSoftVD são condizentes com o que modelos de capacidade de processo existentes no mercado definem como sendo o ideal.

6.4. Análise das características de qualidade do Modelo ProcSoftVD

O modelo ProcSoftVD foi avaliado pelos profissionais (P1, P2, P3, P4, P5 e P6) quanto a cinco características de qualidade definidas pela ISO/IEC 9126-1²⁵, adaptadas para o produto “modelo de processo de software”. As características avaliadas são: funcionalidade, usabilidade, manutenibilidade, portabilidade e eficiência.

Os profissionais selecionaram entre os níveis de atendimento “alto”, “médio”, “baixo” e “NA – não se aplica” do modelo ProcSoftVD para cada uma das subcaracterísticas da característica avaliada e, também, fizeram comentários justificando suas respostas. As opções selecionadas pelos profissionais quanto às subcaracterísticas foram organizadas no Quadro 6.1.

²⁵ ISO/IEC 9126-1: *International Standard Organization (2001). Software Engineering - Product Quality. Part 1: Quality Model.*

QUADRO 6.1 – CARACTERÍSTICAS DE QUALIDADE DO MODELO PROCSTVD

Características	Subcaracterísticas	P1	P2	P3	P4	P5	P6
Funcionalidade: Satisfaz as necessidades?	Adequação: Propõe-se a fazer o que é apropriado?	Alto	Alto	Alto	Médio	Alto	Alto
	Acurácia: Faz o que foi proposto de forma precisa?	Alto	Alto	Médio	Baixo	Médio	Alto
	Interoperabilidade: Interage com modelos/normas especificados?	Alto	Alto	Médio	Médio	Médio	Alto
	Segurança: Evita acesso não autorizado aos dados?	Baixo	NA	NA	NA	NA	Baixo
Usabilidade: É fácil de usar?	Inteligibilidade: É fácil entender sua aplicação e operação?	Alto	Alto	Médio	Baixo	Médio	Alto
	Aprendizagem: É fácil de aprender a usar?	Alto	Médio	Médio	Baixo	Médio	Médio
	Operacionalização: É fácil de operar e controlar?	Alto	Alto	Médio	Baixo	Médio	Médio
	Atratividade: Pode ser atraente ao usuário?	Alto	Alto	Médio	Baixo	Médio	Alto
Manutenibilidade: É fácil de modificar?	Analisabilidade: É fácil de encontrar uma falha, quando ocorre?	Alto	Médio	Baixo	Baixo	Alto	Alto
	Modificabilidade: É fácil modificar e adaptar?	Médio	Alto	Médio	Baixo	Alto	Alto
	Estabilidade: Há grande risco quando se faz alterações?	Alto	Alto	Alto	Baixo	Médio	Médio
	Testabilidade: É fácil testar quando se faz alterações?	Alto	Alto	Médio	Baixo	Médio	Médio
Portabilidade: É fácil de usar em outro ambiente (no caso, em outra MPE)?	Adaptabilidade: É fácil adaptar a outros ambientes (no caso, outras MPEs)?	Médio	Médio	Alto	NA	Médio	Alto
	Capacidade para ser instalado: É fácil instalar em diversos ambientes (no caso, em diversas MPEs)?	Baixo	Alto	Alto	NA	Médio	Alto
	Capacidade para co-existir: É capaz de repartir os recursos com outro modelo de processo de software?	Alto	Médio	Médio	NA	Médio	Alto
	Capacidade para substituir: Ele pode facilmente substituir outro modelo de processo de software?	NA	Alto	Baixo	NA	NA	Alto
Eficiência: É rápido e "enxuto"?	Tempo de resposta: Qual é o tempo de resposta, a velocidade de execução?	Alto	Alto	Baixo	Baixo	Médio	Médio
	Recursos empregados: Quanto recurso usa? Durante quanto tempo?	Baixo	Alto	Alto	Alto	Alto	Alto

Pode-se verificar divergência nas respostas dos profissionais P1, P2, P3, P4, P5 e P6 (Figura 6.2), uma vez que não houve consenso em relação a nenhuma característica analisada, evidenciando a influência dos conhecimentos prévios desses profissionais.

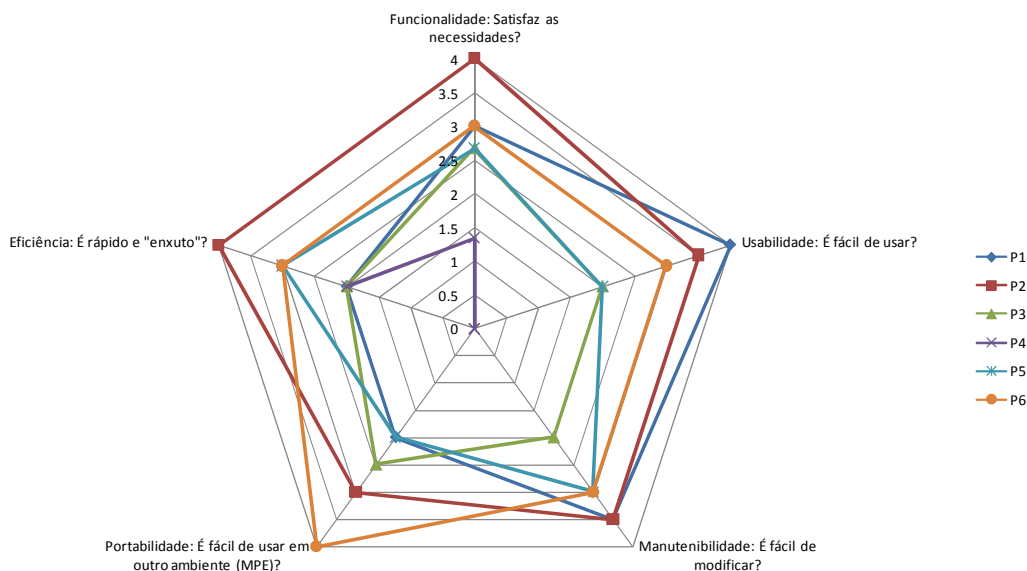


FIGURA 6.2 – GRÁFICO DE RELAÇÃO ENTRE AS RESPOSTAS DOS PROFISSIONAIS

Entretanto, os comentários dos profissionais relacionado a cada uma das subcaracterísticas foram compilados e resultaram nas seguintes informações:

- **Quanto à funcionalidade do modelo ProcSoftVD:** o modelo propõe-se a realizar o que é apropriado, pois oferece um guia de melhoria do processo de software para MPEs com *templates* bem elaborados e sugestivos que conduzem o usuário. Entretanto, a estrutura organizacional da empresa (considerando o conhecimento dos colaboradores e infraestrutura da organização) é um fator determinante para o sucesso da aplicação do modelo.
- **Quanto à usabilidade do modelo ProcSoftVD:** o modelo apresenta as atividades com um nível de esclarecimento, explicações e orientações muito bom e, portanto, fácil de entender. Entretanto, o uso do modelo se torna mais fácil quando o usuário já tem conhecimentos relacionados aos processos de desenvolvimento. A apresentação do modelo em um site permite que o modelo seja consultado por qualquer usuário que navegue pela internet em busca de possíveis soluções para os problemas em sua empresa por falta de um processo bem definido.
- **Quanto à manutenibilidade do modelo ProcSoftVD:** como qualquer outro modelo, o ProcSoftVD apresenta relacionamentos entre as atividades das diversas áreas de conhecimento. Sendo assim, durante o processo de instanciação do modelo em uma MPE, é preciso ter atenção ao serem realizadas as adaptações pertinentes, pois uma modificação em uma atividade pode impactar várias outras atividades do modelo.
- **Quanto à portabilidade do modelo ProcSoftVD:** existe a possibilidade de instanciar todo o modelo para empresa ou instanciar apenas determinadas áreas de conhecimento para ser utilizada em uma MPE. Entretanto, sugere-se que a instância do modelo para a empresa seja realizada por alguém

que tenha conhecimento sobre atividades existentes em modelos de processo de software.

- **Quanto à eficiência do modelo ProcSoftVD:** considerando que a eficiência refere-se à rapidez de aplicação do modelo e também à questão de recursos necessários para essa aplicação, é preciso realizar a implantação do modelo em outras MPEs para se ter uma amostra mais significativa para se chegar a uma conclusão quanto a essa característica. De qualquer modo, é difícil quantificar os recursos a serem usados, pois uma mesma pessoa na empresa (principalmente em MPEs) pode realizar vários papéis. Além disso, instanciar um modelo não é uma tarefa rápida e trivial. Em princípio, acredita-se que enquanto os usuários não estiverem familiarizados com o processo esse tende a ser realizado de maneira lenta.

6.5. Considerações Finais

Neste capítulo foram apresentados os resultados da aplicação do modelo em uma MPE e as avaliações do modelo, segundo seus pontos fortes e fracos e segundo as características de qualidade, por 6 (seis) profissionais da área de computação.

O próximo capítulo discorrerá sobre as conclusões e trabalhos futuros.

7. Conclusões e Trabalhos Futuros

7.1. Considerações Iniciais

O presente trabalho de pesquisa propõe um Modelo de Gestão do Processo de Venda e Desenvolvimento de Software *On-demand* para MPEs (ProcSoftVD - Gestão) que integra os modelos/normas de capacidade de processo CMMI-DEV e ISO/IEC 15504-5 com o framework *Unified Process*, no contexto de gestão de processos de negócio.

A avaliação do modelo foi realizada utilizando uma amostra de seis profissionais da área e duas MPEs, amostra essa considerada pequena dada a complexidade do assunto analisado. Isso evidencia uma restrição metodológica do trabalho, já que não foi possível considerar uma amostra representativa. Assim, as conclusões aqui expostas referem-se a uma visão parcial. Entretanto, essa visão permitiu identificar algumas evidências sobre pontos vulneráveis do modelo que serão expostos (juntamente aos pontos fortes do modelo) na próxima seção, e que deverão ser explorados com maior rigor em trabalhos futuros, descritos na seção 7.3.

7.2. Conclusões

Por meio da análise dos dados obtidos durante as avaliações do modelo, pode-se concluir que para as MPEs analisadas foi mais interessante utilizar como modelo de referência para criar o seu processo padrão o ProcSoftVD, ao invés de criar o seu processo padrão a partir de modelos de capacidade de processo, tais como o CMMI e a ISO/IEC 15504-5, devido ao nível de detalhamento das atividades por meio das tarefas e *templates* sugeridos.

O modelo ProcSoftVD aborda vários aspectos relevantes do processo de desenvolvimento de software para as MPEs. É viável, pois além de descrever cada fase do modelo proposto com um vocabulário de fácil entendimento, ainda

indica/sugere formas para sua aplicação na empresa. Além disso, os artefatos criados pelas atividades possuem referências, fundamentação teórica e exemplos sobre cada assunto que pode não ser trivial ao seu público-alvo.

Mostrou-se útil para conscientizar uma MPE sobre a necessidade e vantagens de se ter um processo padrão, uma vez que atualmente várias empresas estão perdendo licitações por não atenderem aos modelos de capacidade de processo, como o CMMI. Uma certificação desse tipo é bastante cara e talvez uma MPE ainda não tenha recursos o suficiente para tal. Nesse contexto, o ProcSoftVD - Gestão pode ser acessado por qualquer MPE brasileira, pois seu acesso é livre e gratuito, o que minimiza os investimentos com consultoria externa.

Uma das características relevantes do modelo ProcSoftVD é a possibilidade de selecionar as áreas de conhecimento mais prioritárias a serem implantadas na empresa aos poucos, de acordo com os ciclos de melhoria definidos na aplicação do Método de Melhoria de Processo de Venda e Desenvolvimento de Software *on-demand* (ProcSoftVD – Melhoria) elaborado neste trabalho de pesquisa. Essa característica foi evidenciada durante a instanciação do modelo ProcSoftVD nas duas MPEs participantes do projeto de pesquisa.

Uma característica inédita identificada foi a abordagem de atividades da área de conhecimento “Comercialização” (além das sugeridas pela ISO/IEC 15504-5) e as fases de “Prospecção” e “Negociação” no processo de desenvolvimento de software *on-demand*, pois envolvem atividades de busca por potenciais clientes, atividades de entendimento das necessidades do cliente para que haja um acordo entre as partes envolvidas - o cliente e a empresa - em relação aos requisitos a serem atendidos, atividades relacionadas à viabilidade financeira do projeto e relacionadas ao pagamento dos produtos/serviços vendidos pela empresa.

Em relação à área de conhecimento “Gestão de Conhecimento”, não foi possível no âmbito deste trabalho encontrar evidências a respeito de sua efetividade, pois nos ciclos de melhoria definidos pelas MPEs participantes do projeto essa área

não foi considerada nos primeiros ciclos de melhoria. Acompanhar o ciclo completo de melhoria do processo na empresa requer mais tempo de implantação e análise.

7.3. Trabalhos Futuros

Alguns trabalhos futuros propostos, a partir da análise dos resultados e conclusões expostos são:

- Aplicação do modelo em outras MPEs para se ter uma amostra mais significativa;
- Definição de um ambiente (PSEE) que apóie a definição do modelo de processo de uma MPE, a partir do Modelo ProcSoftVD;
- Expansão do modelo para cobertura de outras áreas de conhecimento;
- Definição de perfis de capacidade de processo voltados a possíveis situações enfrentadas por MPE's;
- Análise da inclusão de práticas ágeis no Modelo ProcSoftVD;
- Mapeamento das atividades do Modelo ProcSoftVD com os resultados esperados do MPS.BR;
- Investigação de como utilizar o Modelo ProcSoftVD como material de treinamento para melhoria de processos em MPEs;
- Adaptação do modelo para o caso de incubadoras.

7.4. Considerações Finais

Este capítulo apresentou as conclusões sintetizadas a partir dos resultados desse trabalho de pesquisa, além dos trabalhos que podem ser desenvolvidos na linha de evolução do Modelo ProcSoftVD – Gestão.

Referências

AALST, W. M. P. van der; HOFSTEDE, A. H. M. ter; WESKE, M. (2003). **Business Process Management: A Survey**. BPM Center Report BPM-03-02, BPMcenter.org, 2003. Disponível em: <<http://is.tm.tue.nl/staff/wvdaalst/BPMcenter/reports/2003/BPM-03-02.pdf>>. Acesso em: 20 fev 2007.

AALST, W. M. P. van der (2004). **Business Process Management Demystified: A Tutorial on Models, Systems and Standards for Workflow Management**. In: Desel, J.; Reisig, W.; Rozenberg, G. (Eds): ACPN 2003, LNCS 3098, pp. 1-65, Springer-Verlag Berlin Heidelberg, 2004.

ABRAN, A.; MOORE, J. W.; BOURQUE, P.; DUPUIS, R.; TRIPP, L. L. (2004). Guide to the **Software Engineering Body of Knowledge - SWEBOK**, 2004 version, The Institute of Electrical and Electronics Engineers, Inc. - IEEE, 210 pages.

AGUILAR-SÁVEN, R. S. & OLHAGER, J. (2002). **Integration of product, process and functional orientations: Principals and a case study**. In: *International Conference on Advance Production Management Systems*. The Netherland.

AIR FORCE (1980). **IDEF0 CAM**. Arlington, Texas: AIR FORCE, 1980.

AMBLER, S. W. (2005). **A Manager's Introduction to The Rational Unified Process (RUP)**. Disponível em: <<http://www.ambyssoft.com/downloads/managersIntroToRUP.pdf>>. Acessado em: 26 out 2006.

ARBAQUI, S.; DERNIAME, J. C.; OQUENDO, F.; VERJUS, H. (2002). **A Comparative Review of Process-Centered Software Engineering Environments**. In: *Annals of Software Engineering* 14, 311-340, Kluwer Academic Publishers.

BASIL, V. R.; WEISS, D. M. (1984). **A Methodology for Collecting Valid Software Engineering Data**. In: *IEEE Transactions on Software Engineering*, SE-10(6), 728-738.

BOEHM, B.; EGYED, A.; PORT, D.; SHAH, A.; KWAN, J.; MADACHY, R. (1998). **A stakeholder win-win approach to software engineering education**. *Annals of Software Engineering* 6 (1998) 295-321.

BOGEN, J.; GRONAU, N.; SCHMID, S. (2005). **Improvement of software engineering by modeling of knowledge intensive business processes**, Technical Report, WI 12/2005. Disponível em: <[http://wi.uni-potsdam.de/hp.nsf/0/B0F253DF0812167CC12570180031E56F/\\$FILE/WI-2005-12.pdf](http://wi.uni-potsdam.de/hp.nsf/0/B0F253DF0812167CC12570180031E56F/$FILE/WI-2005-12.pdf)>. Acessado em: 10 nov 2006.

BORSSATTO, I. (2008). **A implementação do MPS.BR nível F na Synos**. Disponível em: <<http://www.softex.br/mpsBr/artigos/artigo.asp?id=1828>>. Acesso em: 05 jun 2008.

BUSINESS PROCESS MANAGEMENT INITIATIVE (2008). **BPMN – Business Process Modeling Notation**. Disponível em: <<http://www.bpmi.org>>. Acesso em: 05 mai 2008.

BRYMAN, A. (1989). **Research Methods and Organization Studies**. Routledge.

- CASSIOLATO, J. E.; LASTRES, H. M. M. (2001). Arranjos e Sistemas Produtivos Locais na Indústria Brasileira. **Revista de Economia Contemporânea**. Rio de Janeiro: UFRJ. Disponível em: <http://www.desenvolvimento.gov.br/arquivo/sti/publicacoes/futAmaDilOportunidades/rev20010424_04.pdf> . Acesso em: 30 set 2005.
- CERVO, A. L. & BERVIAN, P. A. (2002). **Metodologia Científica**. 5ª ed. São Paulo: Pearson Prentice Hall.
- CHRISISS, M. B.; KONRAD, M.; SHRUM, S. (2003). **CMMI: Guidelines for Process Integration and Product Improvement**, Addison-Wesley Pub Co, 2003.
- COSTA, J.M.H. (2006). **Proposta de uma metodologia de gestão de mudanças: aplicação em uma empresa desenvolvedora de software**. 208f. Dissertação (Mestrado) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2006.
- CURTIS, B. (1998). **Which Comes First, the Organization or its Processes**. IEEE Software, pgs 10-13, November/December 1998.
- DANE, F. C. (1990). **Research Methods**. Belmont, California, Brooks/Cole.
- DAVENPORT, T. H. (1993). **Need radical innovation and continuous improvement? Integrate process reengineering and TQM**. Planning Review, 21 (3), p. 6-12.
- DEMING, E. (1986). **Out of the Crisis**. Cambridge. MIT Center for Advanced Engineering Study, 1986.
- DeSOUZA, K. C. (2003). **Barriers to Effective Use of Knowledge Management Systems in Software Engineering**. Communications of the ACM, January, 2003, v. 46, n.1, p. 99-101.
- ECLIPSE FOUNDATION (2008). **Eclipse Process Framework**. Disponível em: <<http://www.eclipse.org/epf/>>. Acesso em: 15 mai 2008.
- ECLIPSE PROCESS FRAMEWORK COMMUNITY (2008). **OpenUP**. Disponível em: <<http://epf.eclipse.org/wikis/openup/>>. Acesso em: 15 mai 2008.
- EL EMAM, K. (2004). **Software Engineering Process**. In: Chapter 9 of Guide to the SoftwareEngineering Body of Knowledge (SWEBOK), 2004 version, (Abran et al. 2004), pgs 9-1 to 9-14. Disponível em: <<http://www.swebok.org/>>. Acesso em: 15 mai 2008.
- FALBO, R. A.; ARANTES, D. O.; NATALI, A. C. C. (2004). **Integrating Knowledge Management and Groupware in a Software Development Environment**. In: D. Karagiannis and U. Reimer (eds.): PAKM 2004, LNAAI 3336, pp. 94-105.
- FIGUEIREDO, S.; SANTOS, G.; MONTONI, M.; ROCHA, A. R.; BARRETO, Andréa.; BARRETO, Ahilton; FERREIRA, A. (2006). **Taba Workkstation: Supporting Technical Solution Through Knowledge Management of Design Rationale**. In: REIMER, U. & KARAGIANNIS, D. (Eds.): PAKM 2006, LNAI 4333, Springer-Verlag Berlin Heidelberg, p. 61-72.
- GIL, A. C. (1999). **Métodos e Técnicas da Pesquisa Social**. 5ª ed. São Paulo: Atlas.

- HUNG, R. Y. (2006). **Business process management as competitive advantage: a review and empirical study**. Total quality management. 17(1), 21–40, jan, 2006.
- IBM. (2006). **Rational Method Composer**. Disponível em: <http://www-128.ibm.com/developerworks/downloads/r/rup/support.html?S_TACT=105AGX28&S_CMP=DLMAIN>. Acesso em: 02 fev 2006.
- INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. (2000). **ISO 9001:2000**. Quality management systems. Requirements.
- INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. (2001). **ISO/IEC 12207 AMD1 e AMD2**: Information technology — Software life cycle processes.
- INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. (2003). **ISO/IEC 15504**: Information Technology – Process Assessment.
- JACOBSON, I.; BOOCH, G.; RUMBAUGH, J. (1999). **The Unified Software development Process**. Addison Wesley.
- JESTON, J.; NELIS, J. (2008). **Business Process Management – Practical Guidelines to Successful Implementations**. 2th edition. Butterworth-Heinemann.
- KILMANN, R. (1995). **A holistic program and critical success factors of corporate transformation**. European Management Journal, 13 (2), p. 175-186.
- KRUCHTEN, P. (2004). **The Rational Unified Process: An Introduction**. Addison-Wesley.
- LIMA, A.; FRANÇA, B.; SCHLEBBE, H.; REIS, R. Q.; REIS, C. L. (2006). **WebAPSEE: Um Ambiente Livre e Flexível para Gerência de Processos de Software**. VII Workshop de Software Livre. Porto Alegre, abr.
- MCFEELEY, B. (1996). **IDEALSM: A User's Guide for Software Process Improvement**. Handbook CMU/SEI-96-HB-001. Disponível em: <<http://www.sei.cmu.edu/pub/documents/96.reports/pdf/hb001.96.pdf>>. Acesso em: 15 mai 2008.
- MCGOWAN, D. A.; MARCA, C. L. (1987). **SADT: Structured Analysis and Design Technique**. McGraw-Hill, New York.
- McKAY, A. & RADNOR, Z. (1998). **A characterization of a business process**. International Journal of Operational and Production Management, 18 (9/10), p. 924-936.
- MCT (2005). **Qualidade e Produtividade no Setor Brasileiro. Pesquisa 2005**. Disponível em: <<http://www.mct.gov.br/index.php/content/view/3253.html>>. Acesso em: 15 mai 2008.
- MICROSOFT. (2006). **Visual Studio Team System**. Disponível em: <<https://ms.helifan.net/brasil/msdn/teamsystem/default.aspx>>. Acesso em: 02 fev 2006.
- MONTEIRO, R. W.; MARTINS, C.; CABRAL, R.; ROCHA, A. R. (2008). **A Empresa de Processamento de Dados do Estado do Pará Rumo ao Nível F do MR-MPS**.

Disponível em: <http://www.softex.br/mpsBr/_artigos/artigo.asp?id=1828>. Acesso em: 05 jun 2008.

MONTONI, M.; MIRANDA, R.; ROCHA, A. R.; TRAVASSOS, G. H. (2004a). **Knowledge Acquisition and Communities of Practice: An Approach to Convert Individual Knowledge into Multi-organizational knowledge**. In: G. Melnik and H. Holz (eds.): LSO 2004, LNCS 3096, pp. 110-121.

MONTONI, M.; SANTOS, G.; VILLELA, K.; MIRANDA, R.; ROCHA, A. R.; TRAVASSOS, G. H.; FIGUEIREDO, S.; MAFRA, S. (2004b). **Knowledge Management in an Enterprise-Oriented Software Development Environment**. In: D. Karagiannis and U. Reimer (eds.): PAKM 2004, LNAI 3336, pp. 117-128.

MONTONI, M.; SANTOS, G.; VILLELA, K.; ROCHA, A. R.; TRAVASSOS, G. H.; FIGUEIREDO, S.; MAFRA, S.; ALBUQUERQUE, A.; MIAN, P. (2005). **Enterprise-Oriented Software Development Environments to Support Software Products and Process Quality Improvement**. In: BOMARIUS, F. & SIRVIO, K. (Eds.): PROFES 2005, LNCS 3547, Springer-Verlag Berlin Heidelberg, p. 370-384.

MONTONI, M.; SANTOS, G.; ROCHA, A. R. WEBER, K. C.; ARAÚJO, E. E. R. (2007). **MPS Model and TABA Workstation: Implementing Software Process Improvement Initiatives in Small Settings**. In: Fifth International Workshop on Software Quality (WoSQ07), IEEE Computer Society.

OLIVEIRA, A. C. G.; GUIMARÃES, F. A.; FONSECA, I. A. (2008). **Utilizando Metodologias Ágeis para Atingir MPS.BR Nível F na PowerLogic**. Disponível em: <http://www.softex.br/mpsBr/_artigos/artigo.asp?id=1828>. Acesso em: 05 jun 2008.

PÁDUA, E. M. M. (1996). **Metodologia da Pesquisa: Abordagem Teórico-Prática**. Campinas-SP: Papyrus.

PAULK, M. C.; WEBER, C. V.; CURTIS, B.; CHRISSIS, M. B. (1995). **The Capability Maturity Model: Guidelines for Improving the Software Process**. SEI Series in Software Engineering, Addison-Wesley.

PINO, F. J.; GARCIA, F.; PIATTINI, M. (2008). **Software Process Improvement in Small and Medium Software Enterprises: a Systematic Review**. In: Software Quality Journal, 16:237-261.

PORTO, J.B.; LÓPEZ, P. A. P.; ALBERTUNI, I.; RICHTER, L. A.; CORRÊA NETO, J. A. (2008). **A Experiência de Avaliação MPS.BR Nível F na Qualidade**. Disponível em: <http://www.softex.br/mpsBr/_artigos/artigo.asp?id=1828>. Acesso em: 05 jun 2008.

PRESSMAN, R. S. (2006). **Engenharia de Software**. 6ª ed. São Paulo: McGraw-Hill.

PROJECT MANAGEMENT INSTITUTE (2003). **Organizational Project Management Maturity Model Handbook (OPM3)**, December 2003.

PROJECT MANAGEMENT INSTITUTE (2004). **A Guide to the Project Management Body of Knowledge, PMBOK Guide, Third Edition**, PMI.

ROCHA, A. R.; MONTONI, M.; SANTOS, G.; MAFRA, S.; FIGUEIREDO, S.; ALBUQUERQUE, A.; MIAN, P. (2005). **Reference Model for Software Process Improvement: A Brazilian Experience**. In: I. Richardson et al. (Eds.): EuroSPI 2005, Lecture Notes in Computer Science (LNCS) 3792, pp. 130-141.

ROCHA, A.R.; MONTONI, M.; SANTOS, G.; OLIVEIRA, K.; NATALI, A. C.; MIAN, P.; CONTE, T.; MAFRA, S.; BARRETO, A.; ALBUQUERQUE, A.; FIGUEIREDO, S.; SOARES, A.; BIANCHI, F.; CABRAL, R.; NETO, A. D. (2006). **Success Factors and Difficulties in Software Process Deployment Experiences based on CMMI and MR-MPS.BR**, In: *Proc. Of the 6th International Workshop on Learning Software Organizations (LSO 2006)*, Rio de Janeiro, Brazil, september, 2006, pp. 77-87.

ROZENFELD, H.; FORCELLINI, F. A.; AMARAL, D. C.; TOLEDO, J. C. de; SILVA, S. L. da; ALLIPRANDINI, D. H.; SCALICE, R. K. (2006). **Gestão de Desenvolvimento de Produtos – Uma referência para a melhoria do processo**. São Paulo: Saraiva.

RUS, I. & LINDVALL, M. (2002). **Knowledge Management in Software Engineering**. IEEE Software, May/Jun, 26-38.

SALVIANO, C. F. (2006). **Uma proposta orientada a perfis de capacidade de processo para evolução da melhoria de processo de software**. Tese de Doutorado sob orientação de Mário Jino. Departamento de Engenharia Elétrica. Universidade Estadual de Campinas, UNICAMP, Brasil.

SCHEER, A.W. (1998). **Business Process Engineering: reference models for industrial enterprises**, Heidelberg, Springer-Verlag.

SCHEID, M.; PESSOA, M. V.; GOMES, R. F.; RAIMUNDO, E. S.; OLIVEIRA, M. A. A. de; SANTOS, G.; FIGUEIREDO, S.; ROCHA, A. R. (2008). **Implantação do MR-MPS Nível E no Centro de Computação da Aeronáutica de São José dos Campos**. Disponível em: <<http://www.softex.br/mpsBr/artigos/artigo.asp?id=1828>>. Acesso em: 05 jun 2008.

SEGRINI, B. M.; BERTOLLO, G.; FALBO, R. A. (2006). **Evoluindo a Definição de Processos de Software em ODE**. In: XIII Sessão de Ferramentas do SBES (Simpósio Brasileiro de Engenharia de Software). Florianópolis/SC, Brasil, p. 109-114.

SMITH, H.; FINGAR, P. (2003). **Business Process Management: The third wave**. Meghan-Kiffer Press, 2003.

SOFTEX (2007). **MPS.BR- Melhoria de Processo do Software Brasileiro . Guia Geral**. Disponível em: <http://www.softex.br/mpsbr/guias/MPS.BR_Guia_Geral_V1.2.pdf>. Acesso em: 15 mai 2008.

SOFTWARE ENGINEERING INSTITUTE (2002). **Capability Maturity Model Integration – CMMI, version 1.1**.

SOFTWARE ENGINEERING INSTITUTE (2006). **CMMI for Development, version 1.2 - CMMI-DEV**. Disponível em: <<http://www.sei.cmu.edu/pub/documents/06.reports/pdf/06tr008.pdf>>. Acesso em: 27 out 2008.

SOMMERVILLE, I. (2007). **Engenharia de Software**. 8^a ed. São Paulo: Pearson Addison Wesley.

SPÍNOLA, R. O.; ÁVILA, A. L. (2008). Introdução à Engenharia de Requisitos. In: Engenharia de Software Magazine, Edição 01, pg 46-54, DevMedia, Disponível em: <<http://www.devmedia.com.br/articles/viewcomp.asp?comp=8034&hl>>. Acesso em: 27 out 2008.

SWEBOK (2004). **SWEBOK – Software Engineering Body of Knowledge**. Disponível em: <<http://www.swebok.org/>>. Acesso em: 08 out 2005.

THIOLLENT, M. (1995). **Metodologia da Pesquisa-Ação**. São Paulo: Cortez.

VARGAS, D.; NIGRI, M.; KRIEGER, M.; BARRETO, A.; MONTONI, M.; CABRAL, R.; ROCHA, A. R. (2008). **Melhoria de Processos na Marlin**. Disponível em: <<http://www.softex.br/mpsBr/artigos/artigo.asp?id=1828>>. Acesso em: 05 jun 2008.

WEBER, K. C.; ARAÚJO, E. R.; ROCHA, A. R.; MACHADO, C. F.; SCALET, D.; SALVIANO, C. F. (2005a). **Brazilian Software Process Reference Model and Assessment Method**, In: P. Yolum et al. (Eds), Proceedings of ISICIS 2005 (20th Int. Symp. On Computer and Info. Sciences), LNCS 3733, pp. 402-411. Copyright Springer-Verlag, Berlin Heidelberg.

WEBER, S.; HAUCK, J. C. R.; VON WANGENHEIM, C. (2005b). **Estabelecendo Processos de Software em Micro e Pequenas Empresas**. SBQS – Simpósio Brasileiro de Qualidade de Software, Porto Alegre, Brasil.

WEBER, K. C.; ARAUJO, E. (2006). **Avaliação do Modelo MPS em Empresas em 2005 e 2006**. Disponível em: <<http://golden.softex.br/portal/softexweb/uploadDocuments/mpsbr/PBQP%202006%20Artigo%20Projeto%204.pdf>>. Acesso em: 15 mai 2008.

WEBER, K. C.; NASCIMENTO, C. J. do; MARINHO, D. S. (2006). **Programa Brasileiro da Qualidade e Produtividade em Software: Treze anos acompanhando e disseminando a cultura da qualidade**. Disponível em: <http://www.mct.gov.br/upd_blob/0006/6446.pdf>. Acesso em: 15 mai 2008.

WELLS, D. (2006). **Extreme Programming: A gentle introduction**. Disponível em: <<http://www.extremeprogramming.org>>. Acesso em: 15 mai 2008.

ZAIRI, M. & SINCLAIR, D. (1995). **BPR and process management: a survey of current practice and future trends in integrated management**. Business Process Re-engineering and Management Journal, 1 (1), p. 8-29.

ZAIRI, M. (1997). **Business process management: a boundaryless approach to modern competitiveness**. Business Process Management Journal, 3 (1), p. 64-80.

Apêndice 1 – Questionário para Levantamento de Áreas de Conhecimento relevantes para algumas MPEs

QUESTIONÁRIO

1. Identificação

1.1 Seu nome (opcional):

1.2 E-mail pessoal para contato:

1.3 Qual é a sua formação acadêmica?

R.

1.4 Qual o nome da faculdade onde se formou? E, em que ano?

R.

1.5 Função que você exerce na empresa:

a. () Gerente/Supervisor

b. () Gerente de Projetos

c. () Analista de Sistemas

d. () Programador

e. () Outra. Qual ?

1.6 Você tem autoridade para fazer alguma modificação na forma de conduzir as atividades de desenvolvimento de software dentro da empresa?

() Sim.

() Não.

2. Caracterização da Empresa de Desenvolvimento de Software ou CPD

2.1 Nome da Empresa/local onde trabalha (opcional):

2.2 Quantidade de funcionários:

() Na empresa

() No CPD da empresa (se for o caso)

2.3 Quanto às atividades de desenvolvimento de software na empresa ou no CPD da empresa (se for o caso), tem-se:

a. () Desenvolvimento de software de prateleira (pacote de software). Ex: software pronto vendido em supermercados.

b. () Desenvolvimento de software sob encomenda (solicitado por um determinado cliente).

c. () Desenvolvimento de software embarcado (que controla outros equipamentos).

- d. () Desenvolvimento de software para internet.
- e. () Desenvolvimento de software para uso próprio.
- f. () Desenvolvimento e distribuição de um único software.
- g. () Outra. Qual?

2.4 Qual(is) o(s) tipo(s) de software(s) desenvolvido(s)?

- () Administração Comercial
- () Administração Escolar
- () Administração Pública
- () Administração de Recursos Humanos
- () Agropecuário
- () Automação Bancária
- () Automação de Escritórios
- () Comércio Eletrônico
- () Contabilidade
- () Educacional
- () Entretenimento
- () Ferramentas de desenvolvimento de sistemas
- () Financeiro
- () Processador de imagens
- () Outro(s). Qual(is)?

2.5 As pessoas que trabalham na equipe de desenvolvimento de software são resistentes às mudanças?

- () Sim.
- () Não.

Comentários:

.....

.....

.....

3. Comercialização do Produto Software

Para quem não assinalou a opção (e.) da questão 2.3, por gentileza, responda às questões de 3.1 a 3.4.

3.1 Quanto à venda do produto, considerando o total de produtos já desenvolvidos em sua empresa:

- a. () Quantos produtos foram vendidos junto ao código-fonte, documentação do produto e manual do usuário?
- b. () Quantos foram vendidos apenas com o código-fonte?
- c. () Quantos foram vendidos apenas com o executável?
- d. () Quantos produtos foram vendidos por licenças?
- e. () Outra situação. Qual?

3.2 O desenvolvimento do sistema é feito por módulos (partes)?

Sim.

Nesse caso, a cobrança ao cliente é feita a cada término do módulo entregue?

Sim.

Não. Como é realizada?

Não. Como é realizada a cobrança?

3.3 É utilizada alguma ferramenta para auxiliar no controle de cobranças?

Sim. Qual(is)?

Não.

3.4 Você acha que seria útil definir algumas atividades e formulários para auxiliar a fase de comercialização do produto de software desenvolvido?

Sim.

Não.

Independente da resposta, por quê?

.....
.....
.....

4. Definição e Construção do Software

4.1 Na empresa onde trabalha, há alguma descrição (seja em documentos ou *site* ou em uma ferramenta/aplicativo utilizado) referente às atividades que devem ser realizadas para a definição do produto a ser desenvolvido?

Sim.

Não.

4.2 Você acha importante entender o negócio do cliente antes de construir um software para que esse atenda aos objetivos do negócio?

Sim.

Não.

4.3 Na empresa onde trabalha, é elaborada alguma modelagem do negócio do cliente para facilitar o entendimento a todos os envolvidos no projeto de desenvolvimento do software?

Sim.

Não.

4.4 Você acha importante que haja algum tipo de modelagem do negócio do cliente para facilitar o entendimento a todos os envolvidos no projeto de desenvolvimento do software?

Sim.

Não.

Independente da resposta, por quê?

.....
.....
.....

4.5 Após realizar o entendimento das necessidades do cliente, vocês realizam alguma atividade anterior à codificação do software, tal como a elaboração de um documento de requisitos e/ou de um modelo de dados?

() Sim. Quais são as atividades?

.....

.....

() Não.

4.6 Quais as ferramentas/aplicativos que vocês usam durante o processo de desenvolvimento de software, considerando desde a atividade de obtenção dos requisitos do cliente até a entrega e/ou implantação do software no cliente e até mesmo o suporte ao cliente?

R:

.....

.....

5. Relevância dos Processos relacionados ao Ciclo de Vida do Software

5.1 Junto ao questionário foi enviada uma planilha denominada “RELEVANCIA_PROCESSOS” onde devem ser selecionadas uma das alternativas disponibilizadas para cada questão.

Nessa planilha estão definidos alguns processos que podem ser utilizados durante o ciclo de vida do software. Na coluna A encontra-se o nome do processo e na coluna F encontra-se uma breve descrição do respectivo processo (para acessar, basta clicar em cima da célula desejada e a descrição aparecerá na barra de fórmulas, como apresentado na Figura A1.1).

	Processos	Grau de Importância	Risco	Realização Atual	Melhoria	Descrição
1						
2	Análise e Resolução de Decisão					o objetivo é analisar as possíveis decisões usando um processo de avaliação formal que avalie as alternativas identificadas em relação aos critérios estabelecidos.
3	Aquisição					o objetivo é analisar as possíveis decisões usando um processo de avaliação formal que avalie as alternativas identificadas em relação aos critérios estabelecidos.
4	Definição do Processo Organizacional					o objetivo é estabelecer e manter um conjunto de processos e padrões para

FIGURA A1.1 – PLANILHA DE RELEVÂNCIA (DESCRIÇÃO DO PROCESSO)

Independente se a sua empresa tem ou não atividades formalizadas em relação ao desenvolvimento do software, defina:

- o grau de importância de cada um dos processos na sua empresa;
- o risco em continuar executando o processo em questão da forma como é executado atualmente.
- se o processo é executado ou não na empresa, atualmente.
- quais processos você escolheria para serem melhorados em sua empresa.

Uma explicação para cada um dos itens anteriores é fornecida como comentário. Para acessar, basta posicionar o mouse em cima da coluna desejada, como mostrado na Figura A1.2.

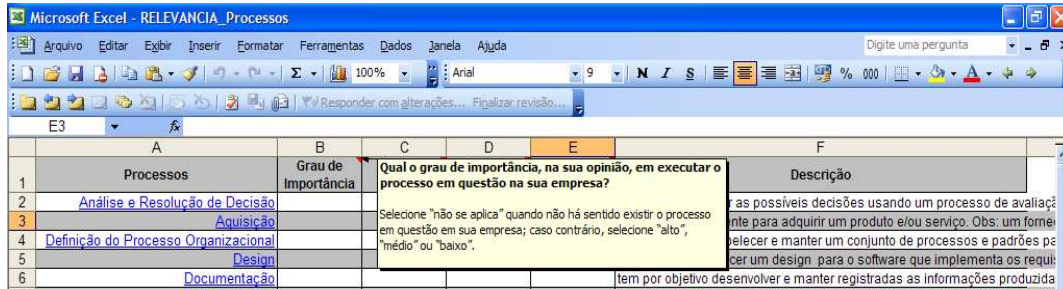



FIGURA A1.2 – PLANILHA DE RELEVÂNCIA (COMENTÁRIO REFERENTE À CADA QUESTÃO)

Para selecionar uma resposta para cada questão, clique na célula correspondente à linha do processo analisado. Aparecerá o ícone , como mostrado na Figura A1.3; Clique nesse ícone para selecionar a opção desejada.

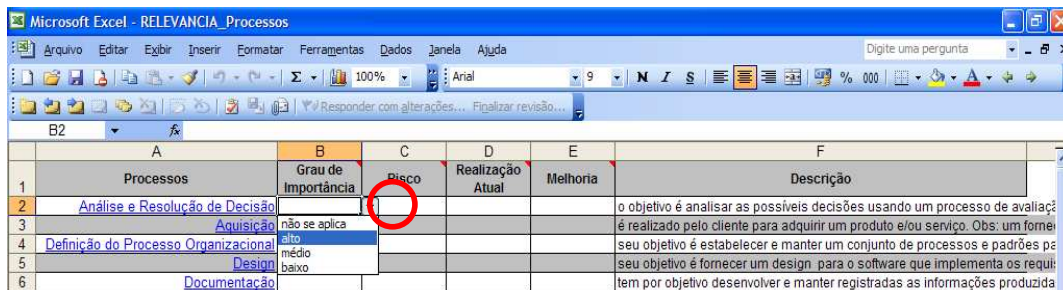


FIGURA A1.3 – PLANILHA DE RELEVÂNCIA (SELEÇÃO DE ALTERNATIVAS)

5.2 Você acredita que um modelo de processo de desenvolvimento de software auxiliaria as pessoas envolvidas no desenvolvimento do software, servindo como um guia comum para todos?

() Sim.

() Não. Por quê?

.....

5.3 Você tem interesse nos resultados dessa pesquisa, depois de finalizada?

() Sim.

() Não.

5.4 Você tem interesse em conhecer o modelo de processo de desenvolvimento de software que será criado nesse trabalho de doutorado?

() Sim.

() Não.

Apêndice 2 – Resultado da Análise do Questionário definido no Apêndice 1

A partir dos questionários respondidos pelas 5 (cinco) MPE's que desenvolvem software e 9 (nove) CPDs (Centro de Processamento de Dados) de empresas diversas os quais desenvolvem software para uso interno, foram confirmadas as áreas de conhecimento definidas no Modelo ProcSoftVD e foi indicada a importância de outras áreas de conhecimento que serão consideradas em um trabalho futuro. A Figura A2.1 apresenta a tabulação das áreas de conhecimento investigadas quanto ao grau de importância de cada uma delas para a empresa e quanto ao grau de risco para a empresa caso não sejam realizadas atividades das respectivas áreas de conhecimento.

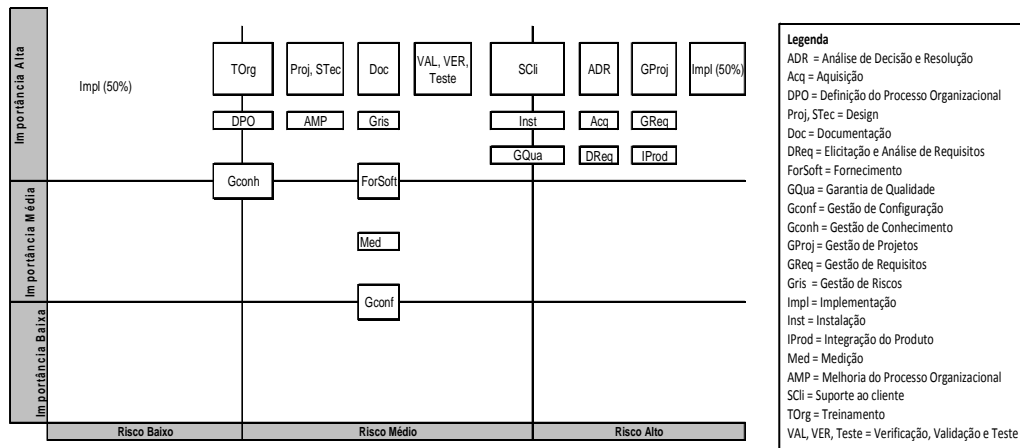


FIGURA A2.1 – TABULAÇÃO DA IMPORTÂNCIA VERSUS RISCO DAS ÁREAS DE CONHECIMENTO PARA A EMPRESA

Percebe-se que para as empresas investigadas as áreas de conhecimento com mais alta prioridade são as que têm alta importância e alto risco caso continue sendo realizada pela empresa da forma que está atualmente. São elas: Gestão de Projetos, Aquisição, Gestão de Requisitos, Elicitação e Análise de Requisitos, Integração do Produto, Análise de Decisão e Resolução. Como segunda prioridade, foram estabelecidas as áreas de conhecimento: Garantia da Qualidade, Instalação e Suporte ao Cliente. Projeto (*design*), Verificação, Validação e Testes foram considerados de

alta importância, porém de risco mediano caso não sejam definidas atividades formalizadas para a empresa.

Apêndice 3 – Modelo ProcSoftVD (Atividades)

Este apêndice apresenta as atividades do modelo ProcSoftVD, agrupadas por fases (Quadros A3.1 a A3.6) e agrupadas por áreas de conhecimento (Quadros A3.7 a A3.19). Além disso, apresenta a visão gráfica do modelo (Figuras A3.20 a A3.26) e o detalhamento de cada uma das atividades (Quadros A3.27 a A3.77).

QUADRO A3.1 – ENTRADAS, ATIVIDADES E SAÍDAS DA FASE PROSPECÇÃO

PROSPECÇÃO: Esta fase é realizada, principalmente, nos casos em que a empresa já tem algum produto/serviço desenvolvido/oferecido ou em fase de desenvolvimento e deseja buscar potenciais clientes para esses tipos de produtos/serviços, a fim de conseguir a solicitação de algumas propostas. Caso não haja a necessidade de prospectar o cliente, realizar as atividades a partir da atividade "P04. Criar infra-estrutura do projeto".

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Plano de Negócios <i>(já deve existir na empresa)</i> Plano de Marketing <i>(já deve existir na empresa)</i> Plano de Ação <i>(já deve existir na empresa)</i> Base de contatos <i>(já deve existir na empresa)</i> 	P01 - Buscar contatos	<ul style="list-style-type: none"> Lista de Contatos
<ul style="list-style-type: none"> Lista de Contatos Plano de Marketing 	P02 - Prospectar cliente	<ul style="list-style-type: none"> Lista de Contatos Roteiro de Prospecção
<ul style="list-style-type: none"> Lista de Contatos 	P03 - Visitar cliente	<ul style="list-style-type: none"> Modelo de Negócio Material de divulgação Lista de Contatos Lista de Requisitos do Cliente
<ul style="list-style-type: none"> Solicitação de Proposta 	P04 - Criar infra-estrutura do projeto	<ul style="list-style-type: none"> Controle de Projetos Proposta do Cliente Impressa
<ul style="list-style-type: none"> Plano de Ação 	P05 - Controlar custos despendidos na fase de prospecção	<ul style="list-style-type: none"> Plano de Ação
Milestone: Solicitação de uma proposta pelo cliente		

QUADRO A3.2 – ENTRADAS, ATIVIDADES E SAÍDAS DA FASE CONCEPÇÃO

CONCEPÇÃO: Esta fase tem como objetivo entender o negócio do cliente, verificar a viabilidade técnica e financeira do projeto para a empresa desenvolvedora, bem como os riscos, ter a visão geral do sistema e sintetizar uma arquitetura candidata para o software. Assim, têm-se subsídios para negociar o contrato com o cliente.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Controle de Projetos Modelo de Negócio Lista de Requisitos do Cliente Relatório de Análise dos Requisitos 	<p>CP01 - Entender o negócio e as necessidades do cliente</p>	<ul style="list-style-type: none"> Modelo de Negócio Lista de Requisitos do Cliente Infra-Estrutura do Cliente Relatório de Análise dos Requisitos
<ul style="list-style-type: none"> Documento de Requisitos 	<p>CP02 - Definir escopo do projeto</p>	<ul style="list-style-type: none"> Plano de Projeto
<ul style="list-style-type: none"> Modelo de Negócio Documento de Requisitos Lista de Requisitos Padrão do Produto (já existente na empresa) 	<p>CP03 - Determinar requisitos do produto</p>	<ul style="list-style-type: none"> Documento de Requisitos Projeto IHM
<ul style="list-style-type: none"> Documento de Requisitos Checklist de Requisitos Modelo de Negócio Projeto IHM Relatório de Análise dos Requisitos 	<p>CP04 - Verificar e Validar os requisitos</p>	<ul style="list-style-type: none"> Relatório de Análise dos Requisitos
<p>Documento de Requisitos</p>	<p>CP05 - Estudar soluções alternativas de arquitetura do software</p>	<ul style="list-style-type: none"> Arquitetura do Software Documento de Requisitos
<ul style="list-style-type: none"> Documento de Requisitos Cronograma Plano de Projeto 	<p>CP06 - Definir/Redefinir cronograma do projeto</p>	<ul style="list-style-type: none"> Cronograma Plano de Projeto
<ul style="list-style-type: none"> Documento de Requisitos Base de Dados Histórica da empresa Cronograma Plano de Projeto 	<p>CP07 - Determinar estimativas de esforço e tempo</p>	<ul style="list-style-type: none"> Cronograma Plano de Projeto

QUADRO A3.2 – ENTRADAS, ATIVIDADES E SAÍDAS DA FASE CONCEPÇÃO (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> • Documento de Requisitos • Plano de Projeto • Arquitetura do Software • Plano de Ação • Infra-Estrutura do Cliente 	<p>CP08 - Verificar a viabilidade técnica e financeira do projeto</p>	<ul style="list-style-type: none"> • Plano de Projeto • Relatório de Viabilidade do Projeto • Fluxo de Caixa
<ul style="list-style-type: none"> • Cronograma • Controle de Projetos • Plano de Projeto 	<p>CP09 - Planejar os recursos do projeto</p>	<ul style="list-style-type: none"> • Plano de Projeto • Controle de Projetos • Pedido de Compra
<ul style="list-style-type: none"> • Controle de Projetos • Plano de Projeto 	<p>CP10 - Identificar conhecimentos e habilidades necessárias</p>	<ul style="list-style-type: none"> • Controle de Projetos • Plano de Projeto
<ul style="list-style-type: none"> • Cronograma • Relatório de Viabilidade do Projeto • Fluxo de Caixa • Plano de Projeto 	<p>CP11 - Identificar riscos inerentes ao projeto</p>	<ul style="list-style-type: none"> • Plano de Projeto
<ul style="list-style-type: none"> • Controle de Projetos • Plano de Projeto 	<p>CP12 - Acompanhar o andamento do projeto</p>	<ul style="list-style-type: none"> • Controle de Projetos • Relatório de Acompanhamento do Projeto
<ul style="list-style-type: none"> • Plano de Ação 	<p>CP13 - Controlar custos despendidos na fase de concepção</p>	<ul style="list-style-type: none"> • Plano de Ação
<p>Milestone: Estudo de Viabilidade do Projeto</p>		
<ul style="list-style-type: none"> • Plano de Garantia da Qualidade (já existente na empresa) • Produto de Trabalho: Relatório de Viabilidade do Projeto 	<p>CP14 - Conduzir revisão do milestone: Estudo de Viabilidade do Projeto</p>	<ul style="list-style-type: none"> • Relatório sobre Garantia da Qualidade

QUADRO A3.3 – ENTRADAS, ATIVIDADES E SAÍDAS DA FASE NEGOCIAÇÃO

NEGOCIAÇÃO: Esta fase tem por objetivo fechar um contrato com o cliente. Para isso, será elaborada, primeiramente, uma proposta pautada no estudo de viabilidade do projeto e no planejamento inicial sob a qual será feita a negociação.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> • Documento de Requisitos • Relatório de Viabilidade do Projeto • Relatório de Análise da Proposta • Cronograma 	N01 - Elaborar a proposta	<ul style="list-style-type: none"> • Proposta Técnica • Proposta Comercial • Relatório de Viabilidade do Projeto • Cronograma • Documento de Requisitos
<ul style="list-style-type: none"> • Proposta Comercial • Proposta Técnica 	N02 - Analisar proposta	<ul style="list-style-type: none"> • Relatório de Análise da Proposta
<ul style="list-style-type: none"> • Proposta • Relatório de Análise do Contrato 	N03 - Elaborar contrato	<ul style="list-style-type: none"> • Contrato
<ul style="list-style-type: none"> • Contrato 	N04 - Analisar contrato	<ul style="list-style-type: none"> • Relatório de Análise do Contrato
<ul style="list-style-type: none"> • Cronograma • Plano de Projeto • Controle de Projetos • Contrato assinado 	N05 - Definir time do projeto	<ul style="list-style-type: none"> • Plano de Projeto • Cronograma • Controle de Projetos
<ul style="list-style-type: none"> • Contrato • Proposta Técnica • Pedido de Compra • Controle de Projetos • Plano de Projeto 	N06 - Formalizar início do projeto	<ul style="list-style-type: none"> • Documento de formalização do projeto • Pedido de Compra • Controle de Projetos
<ul style="list-style-type: none"> • Controle de Projetos • Plano de Projeto • Relatório de Acompanhamento do Projeto 	N07 - Acompanhar o andamento do projeto	<ul style="list-style-type: none"> • Controle de Projetos • Relatório de Acompanhamento do Projeto
<ul style="list-style-type: none"> • Plano de Ação 	N08 - Controlar custos despendidos na fase de negociação	<ul style="list-style-type: none"> • Plano de Ação

Milestone: Contrato

QUADRO A3.3 – ENTRADAS, ATIVIDADES E SAÍDAS DA FASE NEGOCIAÇÃO (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Plano de Garantia da Qualidade Produto de Trabalho: Contrato Relatório sobre Garantia da Qualidade 	<p>N09 - Conduzir revisão do milestone: Contrato</p>	<ul style="list-style-type: none"> Relatório sobre Garantia da Qualidade

QUADRO A3.4 – ENTRADAS, ATIVIDADES E SAÍDAS DA FASE ELABORAÇÃO

ELABORAÇÃO: Esta fase tem o objetivo de definir uma *baseline* da arquitetura do software para fornecer uma base estável para o esforço de implementação na fase de construção e, também, detalhar os requisitos, refinar o cronograma e planejar os testes a serem realizados em cada ciclo de desenvolvimento estimado na fase de concepção.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Documento de Requisitos Projeto IHM Cronograma Modelo de Negócio 	<p>E01 - Detalhar requisitos</p>	<ul style="list-style-type: none"> Documento de Requisitos Projeto IHM Modelo de Negócio
<ul style="list-style-type: none"> Documento de Requisitos Checklist de Requisitos Modelo de Negócio Projeto IHM Relatório de Análise dos Requisitos 	<p>E02 - Verificar e Validar os requisitos</p>	<ul style="list-style-type: none"> Relatório de Análise dos Requisitos
<ul style="list-style-type: none"> Arquitetura do Software Documento de Requisitos Cronograma 	<p>E03- Definir solução de projeto</p>	<ul style="list-style-type: none"> Arquitetura do Software Documento de Requisitos Modelo de Design
<ul style="list-style-type: none"> Cronograma 	<p>E04 - Detalhar o cronograma</p>	<ul style="list-style-type: none"> Controle de Tarefas
<ul style="list-style-type: none"> Plano de Projeto Cronograma Biblioteca de Casos de Teste Padrão Documento de Requisitos Modelo de Design Arquitetura do Software 	<p>E05 - Definir técnicas e critérios para V&V</p>	<ul style="list-style-type: none"> Planilha de Teste Plano de Projeto Cronograma

QUADRO A3.4 – ENTRADAS, ATIVIDADES E SAÍDAS DA FASE ELABORAÇÃO (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Plano de Projeto Relatório de Acompanhamento de Projeto Controle de Projetos Cronograma 	E06 - Monitorar e controlar o projeto	<ul style="list-style-type: none"> Relatório de Acompanhamento de Projeto Controle de Projetos Cronograma
<ul style="list-style-type: none"> Controle de Projetos Relatório de Viabilidade do Projeto Fluxo de caixa 	E07 - Monitorar e Controlar questões financeiras	<ul style="list-style-type: none"> Controle de Projetos Cobrança Fluxo de Caixa
Milestone: Arquitetura do Software		
<ul style="list-style-type: none"> Plano de Garantia da Qualidade Produto de trabalho: Arquitetura do Software Relatório sobre Garantia de Qualidade 	E08 - Conduzir revisão do milestone: Arquitetura do Software	<ul style="list-style-type: none"> Relatório sobre Garantia de Qualidade

QUADRO A3.5 – ENTRADAS, ATIVIDADES E SAÍDAS DA FASE CONSTRUÇÃO

CONSTRUÇÃO: Essa fase tem por objetivo elaborar versões utilizáveis (alpha, beta, e outras entregas testadas) e completar a análise, projeto, desenvolvimento e teste de todas as funcionalidades requeridas.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Arquitetura do Software Modelo de Design Projeto IHM Controle de Tarefas Arquivo de Código Planilha de Teste 	CT01 - Codificar componentes	<ul style="list-style-type: none"> Arquivo de Código Controle de Tarefas
<ul style="list-style-type: none"> Plano de Projeto Planilha de Teste Arquivo de Código 	CT02 - Realizar teste de unidade	<ul style="list-style-type: none"> Planilha de Teste Arquivo de Código
<ul style="list-style-type: none"> Arquivo de Código Planilha de Teste 	CT03 - Analisar os resultados de teste	<ul style="list-style-type: none"> Planilha de Teste
<ul style="list-style-type: none"> Arquivo de Código Plano de Projeto Planilha de Teste 	CT04 - Integrar e testar integração entre componentes	<ul style="list-style-type: none"> Planilha de Teste Arquivo de Código

QUADRO A3.5 – ENTRADAS, ATIVIDADES E SAÍDAS DA FASE CONSTRUÇÃO (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Plano de Projeto Relatório de Acompanhamento de Projeto Controle de Projetos Cronograma 	CT05 - Monitorar e controlar o projeto	<ul style="list-style-type: none"> Relatório de Acompanhamento de Projeto Controle de Projetos Cronograma
<ul style="list-style-type: none"> Controle de Projetos Relatório de Viabilidade do Projeto Fluxo de caixa 	CT06 - Monitorar e Controlar questões financeiras	<ul style="list-style-type: none"> Controle de Projetos Cobrança Fluxo de caixa
Milestone: Capacidade Operacional		
<ul style="list-style-type: none"> Plano de Garantia da Qualidade Produto de Trabalho: Arquivo de Código (Capacidade Operacional do Produto) Relatório sobre Garantia da Qualidade 	CT07 - Conduzir revisão do milestone: Capacidade Operacional	<ul style="list-style-type: none"> Relatório sobre Garantia da Qualidade

QUADRO A3.6 – ENTRADAS, ATIVIDADES E SAÍDAS DA FASE TRANSIÇÃO

TRANSIÇÃO: Essa fase tem por objetivo assegurar que o software esteja disponível para os usuários finais. Essa fase pode atravessar muitas iterações, e inclui testes do produto em preparação para a entrega, e faz os ajustes baseado no *feedback* do usuário. Esses *feedbacks* devem focar principalmente o ajuste do produto, a configuração, instalação e questões de usabilidade.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Arquivo de Código Plano de Projeto Documento de Requisitos Planilha de Teste 	T01 - Validar a versão do produto	<ul style="list-style-type: none"> Planilha de Teste
<ul style="list-style-type: none"> Arquivo de Código Planilha de Teste 	T02 - Analisar e Ajustar a versão do produto	<ul style="list-style-type: none"> Arquivo de Código Planilha de Teste
<ul style="list-style-type: none"> Plano de Projeto Planilha de Teste Arquivo de Código 	T03 - Realizar testes de regressão	<ul style="list-style-type: none"> Planilha de Teste

QUADRO A3.6 – ENTRADAS, ATIVIDADES E SAÍDAS DA FASE TRANSIÇÃO (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Arquivo de Código Manual do Usuário 	T04 - Entregar/Instalar a versão do produto	<ul style="list-style-type: none"> Versão do Produto Documento de Aceite Manual do Usuário Arquivo para Instalação
<ul style="list-style-type: none"> Plano de Projeto Relatório de Acompanhamento de Projeto Controle de Projetos Cronograma 	T05 - Monitorar e controlar o projeto	<ul style="list-style-type: none"> Relatório de Acompanhamento de Projeto Controle de Projetos Cronograma
<ul style="list-style-type: none"> Controle de Projetos Relatório de Viabilidade do Projeto Fluxo de caixa 	T06 - Monitorar e Controlar questões financeiras	<ul style="list-style-type: none"> Controle de Projetos Cobrança Fluxo de caixa
Milestone: Entrega do Produto		
<ul style="list-style-type: none"> Plano de Garantia da Qualidade Produto de Trabalho: Entrega do Produto Relatório sobre Garantia da Qualidade 	T07 - conduzir revisão do milestone: entrega do produto	<ul style="list-style-type: none"> Relatório sobre Garantia da Qualidade

A seguir, os quadros A3.7 a A3.19 apresentam as atividades do modelo ProcSoftVD, com suas respectivas entradas e saídas, agrupadas pelas áreas de conhecimento que compõem o modelo: Comercialização, Modelagem de Negócio, Produção de Requisitos, Projeto, Codificação & Integração de Produto, V&V, Implantação, Aquisição, Medição, Garantia da Qualidade de Produto e Processo, Gestão de Requisitos, Gestão de Mudanças e Configuração, Gestão de Projeto e Gestão do Conhecimento.

QUADRO A3.7 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO COMERCIALIZAÇÃO

COMERCIALIZAÇÃO: essa área de conhecimento foi criada para abordar (1) atividades relacionadas à prospecção de potenciais clientes para a empresa (o que não é coberto pelo CMMI, nem pela ISO/IEC 15504-5); (2) atividades relacionadas à elaboração/submissão de propostas ao cliente e negociação e aprovação de contrato sem ambigüidades que especifique as expectativas, responsabilidades, entregas e compromissos de ambas as partes cliente e fornecedor; (3) atividades relacionadas à viabilidade financeira do projeto e relacionadas ao pagamento dos produtos/serviços vendidos pela empresa. As atividades do item (2) foram definidas com subsídio dos processos “Prospecção do Fornecedor” e “Acordado Contratual” da ISO/IEC 15504-5.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Plano de Negócios <i>(já deve existir na empresa)</i> Plano de Marketing <i>(já deve existir na empresa)</i> Plano de Ação <i>(já deve existir na empresa)</i> Base de contatos <i>(já deve existir na empresa)</i> 	P01 - Buscar contatos	<ul style="list-style-type: none"> Lista de Contatos
<ul style="list-style-type: none"> Lista de Contatos Plano de Marketing 	P02 - Prospectar cliente	<ul style="list-style-type: none"> Lista de Contatos Roteiro de Prospecção
<ul style="list-style-type: none"> Lista de Contatos 	P03 - Visitar cliente	<ul style="list-style-type: none"> Modelo de Negócio Material de divulgação Lista de Contatos
<ul style="list-style-type: none"> Documento de Requisitos Plano de Projeto Arquitetura do Software Fluxo de Caixa Plano de Ação Relatório de Viabilidade do Projeto Infra-Estrutura do Cliente 	CP08 - Verificar a viabilidade técnica e financeira do projeto	<ul style="list-style-type: none"> Plano de Projeto Relatório de Viabilidade Fluxo de Caixa
<ul style="list-style-type: none"> Documento de Requisitos Relatório de Viabilidade do Projeto Relatório de Análise da Proposta Proposta Técnica Proposta Comercial Cronograma 	N01 - Elaborar a proposta	<ul style="list-style-type: none"> Proposta Técnica Proposta Comercial Relatório de Viabilidade do Projeto Cronograma Documento de Requisitos

QUADRO A3.7 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO COMERCIALIZAÇÃO (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Proposta Comercial Proposta Técnica 	N02 - Analisar proposta	<ul style="list-style-type: none"> Proposta Comercial Proposta Técnica Relatório de Análise da Proposta
<ul style="list-style-type: none"> Proposta Relatório de Viabilidade do Projeto Relatório de Análise do Contrato Cronograma 	N03 - Elaborar contrato	<ul style="list-style-type: none"> Contrato
<ul style="list-style-type: none"> Contrato 	N04 - Analisar contrato	<ul style="list-style-type: none"> Relatório de Análise do Contrato Contrato
<ul style="list-style-type: none"> Contrato Proposta Técnica Pedido de Compra Controle de Projetos Plano de Projeto 	N06 - Formalizar início do projeto	<ul style="list-style-type: none"> Documento de formalização do projeto Pedido de Compra Controle de Projetos
<ul style="list-style-type: none"> Contrato Controle de Projetos Relatório de Viabilidade do Projeto 	E07 - Monitorar e Controlar questões financeiras	<ul style="list-style-type: none"> Controle de Projetos Relatório de Viabilidade do Projeto Cobrança Fluxo de Caixa
<ul style="list-style-type: none"> Arquivo de Código Manual do Usuário 	T04 - Entregar/Instalar a versão do produto	<ul style="list-style-type: none"> Versão do Produto Documento de Aceite Manual do Usuário Arquivo para Instalação

QUADRO A3.8 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO MODELAGEM DE NEGÓCIO

MODELAGEM DE NEGÓCIO: o objetivo dessa área de conhecimento é documentar os processos de negócio usando casos de uso de negócio, a fim de assegurar um entendimento comum entre todos os stakeholders (envolvidos) sobre as necessidades existentes no processo de negócio da organização cliente. Essa área de conhecimento teve sua origem no Unified Process.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> • Lista de Contatos 	P03 - Visitar cliente	<ul style="list-style-type: none"> • Modelo de Negócio • Material de divulgação • Lista de Contatos
<ul style="list-style-type: none"> • Controle de Projetos • Modelo de Negócio • Lista de Requisitos do Cliente 	CP01 - Entender o negócio e as necessidades do cliente	<ul style="list-style-type: none"> • Modelo de Negócio • Documento de Requisitos • Infra-Estrutura do Cliente
<ul style="list-style-type: none"> • Documento de Requisitos • Projeto IHM • Cronograma • Modelo de Negócio 	E01 - Detalhar requisitos	<ul style="list-style-type: none"> • Documento de Requisitos • Projeto IHM • Modelo de Negócio

QUADRO A3.9 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO PRODUÇÃO DE REQUISITOS

PRODUÇÃO DE REQUISITOS: tem o objetivo de produzir e analisar os requisitos do cliente, do produto e dos componentes de produto. Essa área de conhecimento teve sua origem relacionada à área de processo “Desenvolvimento de Requisitos” do CMMI-DEV e aos processos “Elicitação de Requisitos”, “Análise de Requisitos do Sistema” e “Análise de Requisitos do Software” da ISO/IEC 15504-5.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> • Lista de Contatos 	P03 - Visitar cliente	<ul style="list-style-type: none"> • Modelo de Negócio • Material de divulgação • Lista de Contatos
<ul style="list-style-type: none"> • Controle de Projetos • Modelo de Negócio • Lista de Requisitos do Cliente 	CP01 - Entender o negócio e as necessidades do cliente	<ul style="list-style-type: none"> • Modelo de Negócio • Documento de Requisitos • Infra-Estrutura do Cliente
<ul style="list-style-type: none"> • Documento de Requisitos 	CP02 - Definir escopo do projeto	<ul style="list-style-type: none"> • Plano de Projeto
<ul style="list-style-type: none"> • Modelo de Negócio • Documento de Requisitos • Lista de Requisitos Padrão do Produto (já existente na empresa) 	CP03 - Determinar requisitos do produto	<ul style="list-style-type: none"> • Documento de Requisitos • Projeto IHM

QUADRO A3.9 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO PRODUÇÃO DE REQUISITOS (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> • Documento de Requisitos • Checklist de Requisitos • Modelo de Negócio 	<p>CP04 - Verificar e Validar os requisitos</p>	<ul style="list-style-type: none"> • Documento de Requisitos • Relatório de Análise dos Requisitos • Modelo de Negócio
<ul style="list-style-type: none"> • Documento de Requisitos 	<p>CP05 - Estudar soluções alternativas de arquitetura do software</p>	<ul style="list-style-type: none"> • Arquitetura do Software • Documento de Requisitos
<ul style="list-style-type: none"> • Cronograma • Relatório de Viabilidade do Projeto • Fluxo de Caixa • Plano de Projeto 	<p>CP11 - Identificar riscos inerentes ao projeto</p>	<ul style="list-style-type: none"> • Plano de Projeto
<ul style="list-style-type: none"> • Documento de Requisitos • Projeto IHM • Cronograma • Modelo de Negócio 	<p>E01 - Detalhar requisitos</p>	<ul style="list-style-type: none"> • Documento de Requisitos • Projeto IHM • Modelo de Negócio
<ul style="list-style-type: none"> • Documento de Requisitos • Checklist de Requisitos • Modelo de Negócio 	<p>E02 - Verificar e Validar os requisitos</p>	<ul style="list-style-type: none"> • Documento de Requisitos • Relatório de Análise dos Requisitos • Modelo de Negócio

QUADRO A3.10 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO PROJETO, CODIFICAÇÃO & INTEGRAÇÃO DE PRODUTO

PROJETO, CODIFICAÇÃO & INTEGRAÇÃO DE PRODUTO: Engloba os processos “Projeto do Software”, “Integração do Software” e “Integração do Sistema” da 15504-5, o workflow “Implementação” do UP e as áreas de processo “Solução Técnica” e “Integração de Produto” do CMMI-DEV. Segundo a 15504-5, o processo “Projeto de Software” tem o objetivo de fornecer um design para o software que é implementado e pode ser verificado em confronto aos requisitos; o processo “Integração de Software” tem o objetivo de combinar as unidades de software, produzindo itens de software integrados, consistentes com o projeto de software, os quais demonstram que os requisitos funcionais e não-funcionais foram satisfeitos; e o processo “Integração do Sistema” tem como objetivo integrar os elementos do sistema (incluindo os itens de software, itens de hardware, operações manuais e outros sistemas, se necessário) para produzir um sistema completo que satisfaça ao projeto do sistema e às expectativas do cliente, expressadas em requisitos do sistema. Segundo o modelo CMMI-DEV, a área de processo “Solução Técnica” tem como objetivo projetar, desenvolver e implementar soluções para os requisitos, soluções essas que envolvem produtos, componentes de produtos e processos de ciclo de vida relacionados ao produto; e a área de processo “Integração do Produto” tem o objetivo de montar o produto, a partir dos componentes de produto, assegurar que o produto ao ser integrado funcione adequadamente, e entregar o produto.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> • Documento de Requisitos 	<p>CP05 - Estudar soluções alternativas de arquitetura do software</p>	<ul style="list-style-type: none"> • Arquitetura do Software • Documento de Requisitos
<ul style="list-style-type: none"> • Arquitetura do Software • Documento de Requisitos • Modelo de Design • Cronograma 	<p>E03- Definir solução de projeto</p>	<ul style="list-style-type: none"> • Arquitetura do Software • Documento de Requisitos • Modelo de Design
<ul style="list-style-type: none"> • Plano de Projeto • Cronograma • Biblioteca de Casos de Teste Padrão • Documento de Requisitos • Modelo de Design • Arquitetura do Software 	<p>E05 - Definir técnicas e critérios para V&V</p>	<ul style="list-style-type: none"> • Planilha de Teste • Plano de Projeto • Cronograma
<ul style="list-style-type: none"> • Arquitetura do Software • Modelo de Design • Projeto IHM • Controle de Tarefas • Arquivo de Código 	<p>CT01 - Codificar componentes</p>	<ul style="list-style-type: none"> • Arquivo de Código • Controle de Tarefas
<ul style="list-style-type: none"> • Arquivo de Código • Planilha de Teste 	<p>T02 - Ajustar a versão do produto</p>	<ul style="list-style-type: none"> • Arquivo de Código
<ul style="list-style-type: none"> • Arquivo de Código • Manual do Usuário 	<p>T04 - Entregar/Instalar a versão do produto</p>	<ul style="list-style-type: none"> • Versão do Produto • Documento de Aceite • Manual do Usuário • Arquivo para Instalação

QUADRO A3.11 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO VERIFICAÇÃO & VALIDAÇÃO

V&V: Engloba as áreas de processo (CMMI) e processos (15504-5) “Validação” e “Verificação” e os processos “Teste de Software” e “Teste de Sistema” da 15504-5. O objetivo da área de processo/processo “Validação” é demonstrar que o produto ou componente do produto atenda ao uso pretendido quando colocado no ambiente destinado. Já o objetivo da área de processo/processo “Verificação” é assegurar que produtos de trabalho (e serviços, no caso do CMMI) selecionados alcancem seus requisitos especificados. Testes desempenham um papel extremamente importante em V&V (Verificação e Validação). Segundo o modelo 15504-5, o processo “Teste” pode ser realizado tanto no software quanto no sistema. O processo “Teste de Software” tem como objetivo confirmar que o produto de software integrado atende aos requisitos definidos. E, o processo “Teste de Sistema” tem o objetivo de assegurar que a implementação de cada requisito de sistema seja testada quanto à sua conformidade e que o sistema esteja pronto para entrega.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> • Documento de Requisitos • Checklist de Requisitos • Modelo de Negócio 	CP04 - Verificar e Validar os requisitos	<ul style="list-style-type: none"> • Documento de Requisitos • Relatório de Análise dos Requisitos • Modelo de Negócio
<ul style="list-style-type: none"> • Documento de Requisitos • Checklist de Requisitos • Modelo de Negócio 	E02 - Verificar e Validar os requisitos	<ul style="list-style-type: none"> • Documento de Requisitos • Relatório de Análise dos Requisitos • Modelo de Negócio
<ul style="list-style-type: none"> • Plano de Projeto • Cronograma • Biblioteca de Casos de Teste Padrão • Documento de Requisitos • Modelo de Design • Arquitetura do Software 	E05 - Definir técnicas e critérios para V&V	<ul style="list-style-type: none"> • Planilha de Teste • Plano de Projeto • Cronograma
<ul style="list-style-type: none"> • Plano de Projeto • Planilha de Teste • Arquivo de Código 	CT02 - Realizar teste de unidade	<ul style="list-style-type: none"> • Planilha de Teste
<ul style="list-style-type: none"> • Arquivo de Código • Planilha de Teste 	CT03 - Analisar os resultados de teste	<ul style="list-style-type: none"> • Planilha de Teste
<ul style="list-style-type: none"> • Arquivo de Código • Plano de Projeto • Planilha de Teste 	CT04 - Integrar e testar integração entre componentes	<ul style="list-style-type: none"> • Planilha de Teste • Arquivo de Código
<ul style="list-style-type: none"> • Arquivo de Código • Plano de Teste • Documento de Requisitos 	T01 - Validar a versão do produto	<ul style="list-style-type: none"> • Planilha de Teste

QUADRO A3.11 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO VERIFICAÇÃO & VALIDAÇÃO (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Arquivo de Código Planilha de Teste 	T02 - Ajustar a versão do produto	<ul style="list-style-type: none"> Arquivo de Código
<ul style="list-style-type: none"> Plano de Projeto Planilha de Teste Arquivo de Código 	T03 - Realizar testes de regressão	<ul style="list-style-type: none"> Planilha de Teste Arquivo de Código

QUADRO A3.12 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO IMPLANTAÇÃO

IMPLANTAÇÃO: o objetivo é entregar o produto produzido para os usuários finais, por meio das atividades de codificação, teste, empacotamento e instalação do software. Essa área de conhecimento teve sua origem no UP e também está relacionada ao processo “Entrega de Produto” da 15504-5.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Arquivo de Código Manual do Usuário 	T04 - Entregar/Instalar a versão do produto	<ul style="list-style-type: none"> Versão do Produto Documento de Aceite Manual do Usuário Arquivo para Instalação

QUADRO A3.13 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO AQUISIÇÃO

AQUISIÇÃO: essa área de conhecimento tem o objetivo de gerenciar a aquisição de produtos de fornecedores, sejam equipamentos ou até mesmo componentes de software do produto (no caso de terceirização do serviço). Exemplos de produtos e componentes de produtos que podem ser adquiridos pelo projeto: subsistemas (por exemplo, um sistema navegacional de uma aeronave), software, hardware, documentação (como manuais de instalação, de operação e do usuário). A origem dessa área de conhecimento está relacionada ao grupo de processo “Aquisição” da 15504-5 à área de processo denominada “Gestão de Acordo com Fornecedor” do CMMI-DEV.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Cronograma Controle de Projetos Plano de Projeto 	CP09 - Planejar os recursos do projeto	<ul style="list-style-type: none"> Plano de Projeto Controle de Projetos Pedido de Compra
<ul style="list-style-type: none"> Pedido de Compras 	ACQ01 - Selecionar fornecedores e estabelecer acordo	<ul style="list-style-type: none"> Produto comprado
<ul style="list-style-type: none"> Pedido de Compras Produto comprado 	ACQ02 - Aceitar o produto adquirido	<ul style="list-style-type: none"> Produto comprado aceito

QUADRO A3.14 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO MEDIÇÃO

MEDIÇÃO: essa área de conhecimento tem o objetivo de coletar e analisar dados relacionados aos produtos desenvolvidos e aos processos implementados dentro da organização por meio de projetos, a fim de dar um suporte efetivo à gestão dos processos e demonstrar objetivamente a qualidade dos produtos. Sua origem está relacionada à área de processo “Análise e Medição” do CMMI-DEV e do processo “Medição” da 15504-5.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> • Documento de Requisitos • Base de Dados Histórica da empresa • Cronograma • Plano de Projeto 	<p>CP07 - Determinar estimativas de esforço e tempo</p>	<ul style="list-style-type: none"> • Cronograma • Documento de Requisitos • Plano de Projeto
<ul style="list-style-type: none"> • Documento de Requisitos • Plano de Projeto • Arquitetura do Software • Fluxo de Caixa • Plano de Ação • Relatório de Viabilidade do Projeto • Infra-Estrutura do Cliente 	<p>CP08 - Verificar a viabilidade técnica e financeira do projeto</p>	<ul style="list-style-type: none"> • Plano de Projeto • Relatório de Viabilidade do Projeto • Fluxo de Caixa
<ul style="list-style-type: none"> • Cronograma • Relatório de Viabilidade do Projeto • Fluxo de Caixa • Plano de Projeto 	<p>CP11 - Identificar riscos inerentes ao projeto</p>	<ul style="list-style-type: none"> • Plano de Projeto
<ul style="list-style-type: none"> • Plano de Projeto • Plano de Garantia da Qualidade (já existente na empresa) • Estudo de Viabilidade do Projeto 	<p>CP14 - Conduzir revisão do milestone: Estudo de Viabilidade do Projeto</p>	<ul style="list-style-type: none"> • Relatório sobre Garantia de Qualidade
<ul style="list-style-type: none"> • Plano de Projeto • Plano de Garantia da Qualidade • Contrato 	<p>N09 - Conduzir revisão do milestone: Contrato</p>	<ul style="list-style-type: none"> • Relatório sobre Garantia de Qualidade

QUADRO A3.14 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO MEDIÇÃO (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Plano de Projeto Plano de Garantia da Qualidade Arquitetura do Software 	E08 - Conduzir revisão do milestone: Arquitetura do Software	<ul style="list-style-type: none"> Relatório sobre Garantia de Qualidade
<ul style="list-style-type: none"> Plano de Projeto Planilha de Teste Arquivo de Código 	CT02 - Realizar teste de unidade	<ul style="list-style-type: none"> Planilha de Teste
<ul style="list-style-type: none"> Arquivo de Código Planilha de Teste 	CT03 - Analisar os resultados de teste	<ul style="list-style-type: none"> Planilha de Teste
<ul style="list-style-type: none"> Arquivo de Código Plano de Projeto Planilha de Teste 	CT04 - Integrar e testar integração entre componentes	<ul style="list-style-type: none"> Planilha de Teste Arquivo de Código
<ul style="list-style-type: none"> Plano de Garantia da Qualidade Produto de Trabalho: Arquivo de Código (Capacidade Operacional do Produto) Relatório sobre Garantia da Qualidade 	CT07 - Conduzir revisão do milestone: Capacidade Operacional	<ul style="list-style-type: none"> Relatório sobre Garantia da Qualidade
<ul style="list-style-type: none"> Arquivo de Código Plano de Teste Documento de Requisitos 	T01 - Validar a versão do produto	<ul style="list-style-type: none"> Planilha de Teste
<ul style="list-style-type: none"> Arquivo de Código Planilha de Teste 	T02 - Analisar e Ajustar a versão do produto	<ul style="list-style-type: none"> Arquivo de Código Planilha de Teste
<ul style="list-style-type: none"> Plano de Projeto Planilha de Teste Arquivo de Código 	T03 - Realizar testes de regressão	<ul style="list-style-type: none"> Planilha de Teste Arquivo de Código

QUADRO A3.14 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO MEDIÇÃO (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Plano de Garantia da Qualidade Produto de Trabalho: Entrega do Produto Relatório sobre Garantia da Qualidade 	T07 - conduzir revisão do milestone: entrega do produto	<ul style="list-style-type: none"> Relatório sobre Garantia da Qualidade
<ul style="list-style-type: none"> Critérios estabelecidos 	Gqpp00 - Desenvolver um Plano de Garantia da Qualidade	<ul style="list-style-type: none"> Plano de Garantia de Qualidade
<ul style="list-style-type: none"> Plano de Garantia de Qualidade 	GqPP01 - Avaliar o processo e assegurar resolução de não conformidades	<ul style="list-style-type: none"> Relatório sobre Garantia da Qualidade
<ul style="list-style-type: none"> Plano de Garantia de Qualidade 	GqPP02 - Avaliar os produtos de trabalho e assegurar resolução de não conformidades	<ul style="list-style-type: none"> Relatório sobre Garantia da Qualidade

QUADRO A3.15 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO GARANTIA DA QUALIDADE DE PRODUTO E PROCESSO

GARANTIA DA QUALIDADE DE PRODUTO E PROCESSO: o objetivo dessa área de conhecimento é fornecer a garantia de que processos e produtos de trabalho estejam em conformidade com planos e provisões pré-definidos. Sua origem está relacionada tanto ao CMMI quanto à ISO/IEC 15504.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Plano de Projeto Plano de Garantia da Qualidade (já existente na empresa) Estudo de Viabilidade do Projeto 	CP14 - Conduzir revisão do milestone: Estudo de Viabilidade do Projeto	<ul style="list-style-type: none"> Relatório sobre Garantia de Qualidade
<ul style="list-style-type: none"> Plano de Projeto Plano de Garantia da Qualidade Contrato 	N09 - Conduzir revisão do milestone: Contrato	<ul style="list-style-type: none"> Relatório sobre Garantia de Qualidade
<ul style="list-style-type: none"> Plano de Projeto Plano de Garantia da Qualidade Arquitetura do Software 	E08 - Conduzir revisão do milestone: Arquitetura do Software	<ul style="list-style-type: none"> Relatório sobre Garantia de Qualidade

QUADRO A3.15 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO GARANTIA DA QUALIDADE DE PRODUTO E PROCESSO (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> • Plano de Garantia da Qualidade • Produto de Trabalho: Arquivo de Código (Capacidade Operacional do Produto) • Relatório sobre Garantia da Qualidade 	<p>CT07 - Conduzir revisão do milestone: Capacidade Operacional</p>	<ul style="list-style-type: none"> • Relatório sobre Garantia da Qualidade
<ul style="list-style-type: none"> • Plano de Garantia da Qualidade • Produto de Trabalho: Entrega do Produto • Relatório sobre Garantia da Qualidade 	<p>T07 - conduzir revisão do milestone: entrega do produto</p>	<ul style="list-style-type: none"> • Relatório sobre Garantia da Qualidade
<ul style="list-style-type: none"> • Plano de Garantia de Qualidade 	<p>GqPP01 - Avaliar o processo e assegurar resolução de não conformidades</p>	<ul style="list-style-type: none"> • Relatório sobre Garantia da Qualidade
<ul style="list-style-type: none"> • Plano de Garantia de Qualidade 	<p>GqPP02 - Avaliar os produtos de trabalho e assegurar resolução de não conformidades</p>	<ul style="list-style-type: none"> • Relatório sobre Garantia da Qualidade

QUADRO A3.16 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO GESTÃO DE REQUISITOS

GESTÃO DE REQUISITOS: o objetivo dessa área de conhecimento, que teve sua origem no CMMI-DEV, é gerenciar os requisitos dos produtos e componentes de produtos dos projetos e identificar inconsistências entre esses requisitos e os planos e produtos de trabalho dos projetos. Para isso, essa área trata do rastreamento dos requisitos em meio ao projeto e das mudanças desses requisitos. As mudanças relacionadas aos requisitos utilizam algumas das atividades definidas pela área de conhecimento “Gestão de Mudanças e Configuração”.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> • Documento de Requisitos • Projeto IHM • Cronograma • Modelo de Negócio 	E01 - Detalhar requisitos	<ul style="list-style-type: none"> • Documento de Requisitos • Projeto IHM • Modelo de Negócio
<ul style="list-style-type: none"> • Mudança interna ou externa • Plano de Gestão de Configuração 	GCf03 - Solicitar mudança(s)	<ul style="list-style-type: none"> • Banco de Dados de Configuração
<ul style="list-style-type: none"> • Banco de Dados de Configuração 	GCf04. Analisar/Autorizar e Planejar mudança(s)	<ul style="list-style-type: none"> • Banco de Dados de Configuração
<ul style="list-style-type: none"> • Plano de Gestão de Configuração • Banco de Dados de Configuração • Itens de configuração afetados 	GCf05 - Implementar mudança(s)	<ul style="list-style-type: none"> • Banco de Dados de Configuração • Itens de configuração afetados

QUADRO A3.17 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO GESTÃO DE MUDANÇAS E CONFIGURAÇÃO

GESTÃO DE MUDANÇAS E CONFIGURAÇÃO: essa área de conhecimento engloba a área de processo “Gestão de Configuração” do CMMI-DEV e os processos “Gestão de Configuração” e “Gestão de Solicitação de Mudanças” da 15504-5. O objetivo dessa área de conhecimento é estabelecer e manter a integridade de produtos de trabalho usando a identificação de configuração, controle de configuração, prestação de contas (explicação) do status da configuração e auditoria da configuração, além de assegurar que solicitações de mudanças no projeto sejam gerenciadas, rastreadas e controladas.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> • Critérios estabelecidos 	Gqpp00 - Desenvolver um Plano de Gestão de Configuração	<ul style="list-style-type: none"> • Plano de Gestão de Configuração
<ul style="list-style-type: none"> • Templates das atividades correspondentes • Plano de Gestão de Configuração 	GCf01 - Criar e identificar os itens de configuração	<ul style="list-style-type: none"> • Itens de configuração identificados • Banco de Dados de Configuração

QUADRO A3.17 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO GESTÃO DE MUDANÇAS E CONFIGURAÇÃO (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Itens de configuração identificados Banco de Dados de Configuração 	GCf02 - Criar e liberar as baselines	<ul style="list-style-type: none"> Baseline Banco de Dados de Configuração
<ul style="list-style-type: none"> Mudança interna ou externa Plano de Gestão de Configuração 	GCf03 - Solicitar mudança(s)	<ul style="list-style-type: none"> Banco de Dados de Configuração
<ul style="list-style-type: none"> Banco de Dados de Configuração 	GCf04. Analisar/Autorizar e Planejar mudança(s)	<ul style="list-style-type: none"> Banco de Dados de Configuração
<ul style="list-style-type: none"> Plano de Gestão de Configuração Banco de Dados de Configuração Itens de configuração afetados 	GCf05 - Implementar mudança(s)	<ul style="list-style-type: none"> Banco de Dados de Configuração Itens de configuração afetados
<ul style="list-style-type: none"> Banco de Dados de Configuração Itens de configuração afetados 	GCf06 - Verificar/Validar e Liberar mudança(s)	<ul style="list-style-type: none"> Banco de Dados de Configuração Itens de configuração afetados

QUADRO A3.18 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO GESTÃO DE PROJETOS

GESTÃO DE PROJETOS: o objetivo dessa área de processo é identificar, estabelecer, coordenar e monitorar as atividades, tarefas e recursos necessários para um projeto produzir um produto e/ou serviço, no contexto dos requisitos e restrições de projetos. Engloba tanto o processo “Gestão de Projetos” da 15504-5 quanto às áreas de processo “Planejamento de Projeto” e “Monitoramento e Controle de Projeto” do CMMI.

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Critérios estabelecidos 	Gcf00 - Desenvolver um Plano de Gestão de Configuração	<ul style="list-style-type: none"> Plano de Gestão de Configuração
<ul style="list-style-type: none"> Critérios estabelecidos 	Gqpp00 - Desenvolver um Plano de Garantia da Qualidade	<ul style="list-style-type: none"> Plano de Garantia de Qualidade

QUADRO A3.18 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO GESTÃO DE PROJETOS (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
	Gco00 - Estabelecer um sistema e estratégia para gestão do conhecimento	<ul style="list-style-type: none"> Sistema de Gestão de Conhecimento
<ul style="list-style-type: none"> Solicitação de Proposta 	P04 - Criar infra-estrutura do projeto	<ul style="list-style-type: none"> Controle de Projetos
<ul style="list-style-type: none"> Documento de Requisitos 	CP02 - Definir escopo do projeto	<ul style="list-style-type: none"> Plano de Projeto
<ul style="list-style-type: none"> Documento de Requisitos Cronograma 	CP06 - Definir/Redefinir cronograma do projeto	<ul style="list-style-type: none"> Cronograma
<ul style="list-style-type: none"> Documento de Requisitos Base de Dados Histórica da empresa Cronograma Plano de Projeto 	CP07 - Determinar estimativas de esforço e tempo	<ul style="list-style-type: none"> Cronograma Documento de Requisitos Plano de Projeto
<ul style="list-style-type: none"> Documento de Requisitos Plano de Projeto Arquitetura do Software Fluxo de Caixa Plano de Ação Relatório de Viabilidade do Projeto Infra-Estrutura do Cliente 	CP08 - Verificar a viabilidade técnica e financeira do projeto	<ul style="list-style-type: none"> Plano de Projeto Relatório de Viabilidade do Projeto Fluxo de Caixa
<ul style="list-style-type: none"> Cronograma Controle de Projetos Plano de Projeto 	CP09 - Planejar os recursos do projeto	<ul style="list-style-type: none"> Plano de Projeto Controle de Projetos Pedido de Compra
<ul style="list-style-type: none"> Controle de Projetos Plano de Projeto 	CP10 - Planejar conhecimentos e habilidades necessárias	<ul style="list-style-type: none"> Controle de Projetos Plano de Projeto

QUADRO A3.18 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO GESTÃO DE PROJETOS (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> • Cronograma • Relatório de Viabilidade do Projeto • Fluxo de Caixa • Plano de Projeto 	<p>CP11 - Identificar riscos inerentes ao projeto</p>	<ul style="list-style-type: none"> • Plano de Projeto
<ul style="list-style-type: none"> • Controle de Projetos • Plano de Projeto • Relatório de Acompanhamento do Projeto 	<p>CP12 - Acompanhar o andamento do projeto</p>	<ul style="list-style-type: none"> • Controle de Projetos • Relatório de Acompanhamento do Projeto
<ul style="list-style-type: none"> • Plano de Ação 	<p>CP13 - Controlar custos dispendidos nessa fase</p>	<ul style="list-style-type: none"> • Plano de Ação
<ul style="list-style-type: none"> • Plano de Projeto • Plano de Garantia da Qualidade (já existente na empresa) • Estudo de Viabilidade do Projeto 	<p>CP14 - Conduzir revisão do milestone: Estudo de Viabilidade do Projeto</p>	<ul style="list-style-type: none"> • Relatório sobre Garantia de Qualidade
<ul style="list-style-type: none"> • Documento de Requisitos • Relatório de Viabilidade do Projeto • Relatório de Análise da Proposta • Proposta Técnica • Proposta Comercial • Cronograma 	<p>N01 - Elaborar a proposta</p>	<ul style="list-style-type: none"> • Proposta Técnica • Proposta Comercial • Relatório de Viabilidade do Projeto • Cronograma • Documento de Requisitos
<ul style="list-style-type: none"> • Cronograma • Plano de Projeto • Controle de Projetos 	<p>N05 - Definir time do projeto</p>	<ul style="list-style-type: none"> • Plano de Projeto • Cronograma
<ul style="list-style-type: none"> • Contrato • Proposta Técnica • Pedido de Compra • Controle de Projetos • Plano de Projeto 	<p>N06 - Formalizar início do projeto</p>	<ul style="list-style-type: none"> • Documento de formalização do projeto • Pedido de Compra • Controle de Projetos

QUADRO A3.18 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO GESTÃO DE PROJETOS (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> • Controle de Projetos • Plano de Projeto • Relatório de Acompanhamento do Projeto 	N07 - Acompanhar o andamento do projeto	<ul style="list-style-type: none"> • Controle de Projetos • Relatório de Acompanhamento do Projeto
<ul style="list-style-type: none"> • Plano de Ação 	N08 - Controlar custos dispendidos nessa fase	<ul style="list-style-type: none"> • Plano de Ação
<ul style="list-style-type: none"> • Plano de Projeto • Plano de Garantia da Qualidade • Contrato 	N09 - Conduzir revisão do milestone: Contrato	<ul style="list-style-type: none"> • Relatório sobre Garantia de Qualidade
<ul style="list-style-type: none"> • Cronograma • Controle de Tarefas 	E04 - Detalhar o cronograma	<ul style="list-style-type: none"> • Controle de Tarefas
<ul style="list-style-type: none"> • Plano de Projeto • Cronograma • Biblioteca de Casos de Teste Padrão • Documento de Requisitos • Modelo de Design • Arquitetura do Software 	E05 - Definir técnicas e critérios para V&V	<ul style="list-style-type: none"> • Planilha de Teste • Plano de Projeto • Cronograma
<ul style="list-style-type: none"> • Plano de Projeto • Relatório de Acompanhamento de Projeto • Controle de Projetos • Cronograma 	E06 - Monitorar e controlar o projeto	<ul style="list-style-type: none"> • Relatório de Acompanhamento de Projeto • Controle de Projetos • Cronograma
<ul style="list-style-type: none"> • Contrato • Controle de Projetos • Relatório de Viabilidade do Projeto 	E07 - Monitorar e Controlar questões financeiras	<ul style="list-style-type: none"> • Controle de Projetos • Relatório de Viabilidade do Projeto • Cobrança • Fluxo de Caixa

QUADRO A3.18 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO GESTÃO DE PROJETOS (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Plano de Projeto Relatório de Garantia da Qualidade Arquitetura do Software 	<p>E08 - Conduzir revisão do milestone: Arquitetura do Software</p>	<ul style="list-style-type: none"> Relatório sobre Garantia de Qualidade
<ul style="list-style-type: none"> Plano de Projeto Relatório de Acompanhamento de Projeto Controle de Projetos Cronograma 	<p>CT05 - Monitorar e controlar o projeto</p>	<ul style="list-style-type: none"> Relatório de Acompanhamento de Projeto Controle de Projetos Cronograma
<ul style="list-style-type: none"> Controle de Projetos Relatório de Viabilidade do Projeto Fluxo de caixa 	<p>CT06 - Monitorar e Controlar questões financeiras</p>	<ul style="list-style-type: none"> Controle de Projetos Cobrança Fluxo de caixa
<ul style="list-style-type: none"> Plano de Garantia da Qualidade Produto de Trabalho: Arquivo de Código (Capacidade Operacional do Produto) Relatório sobre Garantia da Qualidade 	<p>CT07 - Conduzir revisão do milestone: Capacidade Operacional</p>	<ul style="list-style-type: none"> Relatório sobre Garantia da Qualidade
<ul style="list-style-type: none"> Plano de Projeto Relatório de Acompanhamento de Projeto Controle de Projetos Cronograma 	<p>T05 - Monitorar e controlar o projeto</p>	<ul style="list-style-type: none"> Relatório de Acompanhamento de Projeto Controle de Projetos Cronograma
<ul style="list-style-type: none"> Controle de Projetos Relatório de Viabilidade do Projeto Fluxo de caixa 	<p>T06 - Monitorar e Controlar questões financeiras</p>	<ul style="list-style-type: none"> Controle de Projetos Cobrança Fluxo de caixa

QUADRO A3.18 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO GESTÃO DE PROJETOS (CONT.)

ENTRADAS	ATIVIDADES	SAÍDAS
<ul style="list-style-type: none"> Plano de Garantia da Qualidade Produto de Trabalho: Entrega do Produto Relatório sobre Garantia da Qualidade 	<p>T07 - conduzir revisão do milestone: entrega do produto</p>	<ul style="list-style-type: none"> Relatório sobre Garantia da Qualidade

QUADRO A3.19 – ENTRADAS, ATIVIDADES E SAÍDAS DA ÁREA DE CONHECIMENTO GESTÃO DE CONHECIMENTO

GESTÃO DE CONHECIMENTO: tem como objetivo assegurar que o conhecimento, informação e habilidades individuais sejam coletadas, compartilhadas, reusadas e melhoradas por toda a organização. Sua origem está relacionada ao processo “Gestão de Conhecimento” da ISO/IEC 15504.

ENTRADAS	ATIVIDADES	SAÍDAS
	<p>Gco00 - Estabelecer um sistema e estratégia para gestão do conhecimento</p>	<ul style="list-style-type: none"> Sistema de Gestão de Conhecimento
<ul style="list-style-type: none"> Conhecimento Sistema de Gestão de Conhecimento 	<p>GCo01 - Capturar o conhecimento</p>	<ul style="list-style-type: none"> Lições aprendidas e conhecimento
<ul style="list-style-type: none"> Lições aprendidas e conhecimento Sistema de Gestão de Conhecimento 	<p>GCo02 - Lapidar o conhecimento</p>	<ul style="list-style-type: none"> Memória Organizacional
<ul style="list-style-type: none"> Memória Organizacional 	<p>GCo03 - Disseminar o conhecimento</p>	<ul style="list-style-type: none"> Memória Organizacional

As Figuras A3.20 a A3.25, a seguir, apresentam a visão gráfica das atividades em fases. Os Quadros A3.20.1 a A3.20.5 apresentam o detalhamento de cada uma das atividades (responsável, descrição, recursos, entradas, saídas e tarefas) da fase de prospecção. Os Quadros A3.21.1 a A3.21.14 o detalhamento das atividades da fase de concepção, os Quadros A3.22.1 a A.3.22.9 da fase de negociação, os Quadros A3.23.1 a A3.23.8 da fase de elaboração, os Quadros A3.24.1 a A3.24.7 da fase de construção e os Quadros A3.25.1 a A3.25.7 da fase de transição.

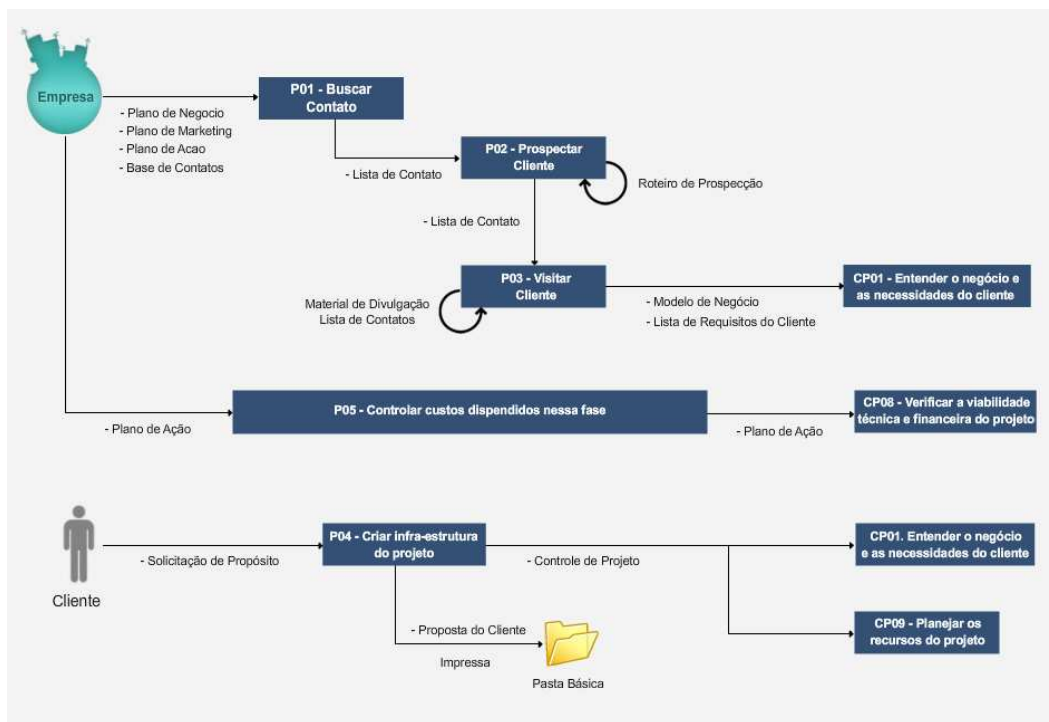


FIGURA A3.20 – VISÃO GRÁFICA – FASE: PROSPECÇÃO

QUADRO A3.20.1 – DETALHAMENTO DA ATIVIDADE P01

Atividade	P01. Buscar contatos
Responsáveis	Equipe de Vendas
Descrição	Essa atividade tem como objetivo buscar na internet ou na base de contatos da empresa os potenciais clientes, de acordo com os perfis estabelecidos no plano de marketing e na estratégia da empresa para o período em questão.
Recursos	- Ferramentas de relacionamento com cliente (CRM), tais como: VTIGER, DBMKT, newsletter, e-mail marketing, mecanismos de buscas, websites
Entradas	e1) Plano de Negócios (já deve existir na empresa) e2) Plano de Marketing (já deve existir na empresa) e3) Plano de Ação (já deve existir na empresa) e4) Base de contatos (já deve existir na empresa)
Saídas	s1) Lista de Contatos
Tarefas	- Executar as atividades para busca de potenciais clientes (item 3.2 do "Plano de Marketing"). * Executar a etapa 1 definida no item "Plano de Ação" (item 4 do "Plano de Marketing"), a fim de validar os contatos, com base no planejamento definido na planilha "Plano de Ação" do arquivo "Plano de Ação". * Criar planilha Lista de Contatos (conforme GCf01) e registrar os contatos validados na planilha "Lista de Contatos", a partir da "Base de Contatos" da empresa. * Registrar a data de realização da etapa 1 descrita no item 4 do "Plano de Marketing" na planilha "Controle de Visitas" do arquivo "Lista de Contatos", para todos os contatos validados.

QUADRO A3.20.2 – DETALHAMENTO DA ATIVIDADE P02

Atividade	P02. Prospectar cliente
Responsáveis	Equipe de Vendas
Descrição	Essa atividade tem como objetivo fazer com que a empresa desenvolvedora entre em contato com o cliente para averiguar a possibilidade de um projeto.
Recursos	- Ferramentas de relacionamento com cliente (CRM), tais como: VTIGER, DBMKT, newsletter, e-mail marketing, mecanismos de buscas, websites
Entradas	e1) Lista de Contatos e2) Plano de Marketing
Saídas	s1) Lista de Contatos s2) Roteiro de Prospecção
Tarefas	<ul style="list-style-type: none"> - Selecionar abordagem do cliente, priorizando contatos que demonstraram interesse depois de executada a etapa 1 do item 4 do "Plano de Marketing". Exemplo de abordagem selecionada: etapa 2 do plano de ação - Contato telefônico - Identificar contato chave para processo de prospecção e registrar na planilha "Lista de Contatos" do arquivo "Lista de Contatos". Exemplo: Contato que tem poder de decisão - Levantar informações sobre o cliente e registrar na planilha "Lista de Contatos" do arquivo "Lista de Contatos". Exemplo: Verificação de gargalos e necessidades, nível de satisfação com o atual prestador de serviços, etc - Criar "Roteiro de Prospecção" (conforme GCf01), de acordo com o item 4 (plano de ação) do "Plano de Marketing" - Entrar em contato com o cliente utilizando o roteiro de prospecção - Registrar a data de comunicação com o cliente na coluna "Etapa 2" na "Lista de Contatos" - Verificar o interesse do cliente por uma proposta e registrar na coluna "Interesse por proposta"

QUADRO A3.20.3 – DETALHAMENTO DA ATIVIDADE P03

Atividade	P03. Visitar cliente
Responsáveis	Equipe de Vendas Cliente
Descrição	Essa atividade tem como objetivo iniciar um relacionamento personalizado com o cliente, a fim de averiguar o interesse do cliente na solicitação de uma proposta de solução para as demandas colocadas.
Recursos	- Meios de deslocamento - Laptop - Ferramenta de modelagem de negócio (ex: Rational Rose, ARPO)
Entradas	e1) Lista de Contatos
Saídas	s1) Modelo de Negócio s2) Material de divulgação s3) Lista de Contatos s4) Lista de Requisitos do Cliente
Tarefas	<ul style="list-style-type: none"> - Pesquisar informações sobre a atividade do cliente e de soluções utilizadas pelo mesmo e registrar na planilha "Lista de Contatos" do arquivo "Lista de Contatos" - Preparar material de divulgação, no caso de existir na empresa algum produto que se encaixe na necessidade do cliente <ul style="list-style-type: none"> * Levantar situações onde a empresa desenvolvedora teve sucesso no atendimento de clientes com problemas análogos * Preparar demonstrações (.ppt, flash) sobre as situações análogas levantadas - Na reunião com o cliente, levantar possíveis demandas, necessidades e informações do cliente, inclusive os casos de uso de negócio do cliente, caso necessário. <ul style="list-style-type: none"> * Criar arquivo "Lista de Requisitos do Cliente" (conforme GCf01) para armazenar as demandas e necessidades do cliente * Solicitar ao cliente que a empresa seja procurada quando surgirem demandas futuras. * Cadastrar a data da visita na planilha "Controle de Visitas" do arquivo "Lista de Contatos" * Criar o arquivo "Modelo de Negócio" (conforme GCf01) e registrar as informações obtidas relacionadas ao vocabulário do negócio (item 1), aos casos de uso de negócio (item 2) e às regras de negócio (item 3)

QUADRO A3.20.4 – DETALHAMENTO DA ATIVIDADE P04

Atividade	P04. Criar infra-estrutura do projeto
Responsáveis	Gerente de Projeto
Descrição	A qualquer momento, um cliente pode solicitar uma proposta e, nesse caso, deve-se criar a infra-estrutura necessária para dar início ao projeto.
Recursos	Ferramenta de gestão de projetos (ex: dotProject, MS-Project, Primavera)
Entradas	e1) Solicitação de Proposta
Saídas	s1) Controle de Projetos s2) Proposta do Cliente Impressa
Tarefas	<ul style="list-style-type: none"> - Caso o arquivo "Controle de Projetos" ainda não tenha sido criado, criá-lo (conforme GCf01) - Definir o ID do projeto (há uma sugestão no item 4 "Estratégia de identificação dos itens de configuração" do "Plano de Gestão de Configuração") - Criar projeto na planilha "Identificação de Projetos" do arquivo "Controle de Projetos" da empresa e preencher os campos: id_projeto e descrição - Criar pasta física e pasta lógica e marcar "sim" nas respectivas colunas da planilha "Identificação de Projetos" do arquivo "Controle de Projetos" * Quanto à proposta solicitada pelo cliente, se for o caso: <ul style="list-style-type: none"> - Registrar data de solicitação da proposta pelo cliente, caso ocorra, na planilha "Controle de Visitas" do arquivo "Lista de Contatos" - Digitalizar a proposta solicitada pelo cliente - Anexar a proposta na pasta lógica - Imprimir a proposta e solicitar que o cliente a assine para que seja armazenada na pasta física. - Selecionar projeto na planilha "Acompanhamento" do arquivo "Controle de Projetos" da empresa e preencher os campos: situação (selecionar a opção "inicializado" do combo-box), data de inicialização do projeto, marcar a fase de concepção, ciclo de desenvolvimento e data de início da fase de concepção

QUADRO A3.20.5 – DETALHAMENTO DA ATIVIDADE P05

Atividade	P05. Controlar custos despendidos na fase de prospecção
Responsáveis	Equipe de Vendas
Descrição	Essa atividade tem como objetivo registrar todos os custos referentes à execução das atividades dessa fase.
Recursos	
Entradas	e1) Plano de Ação
Saídas	s1) Plano de Ação
Tarefas	<ul style="list-style-type: none">- Atualizar na planilha "Acompanhamento" (item "Atividades") do arquivo "Plano de Ação" a quantidade de atividades realizadas de fato- Atualizar na planilha "Acompanhamento" (item "Custos") os custos reais referentes à execução de todas as atividades dessa fase

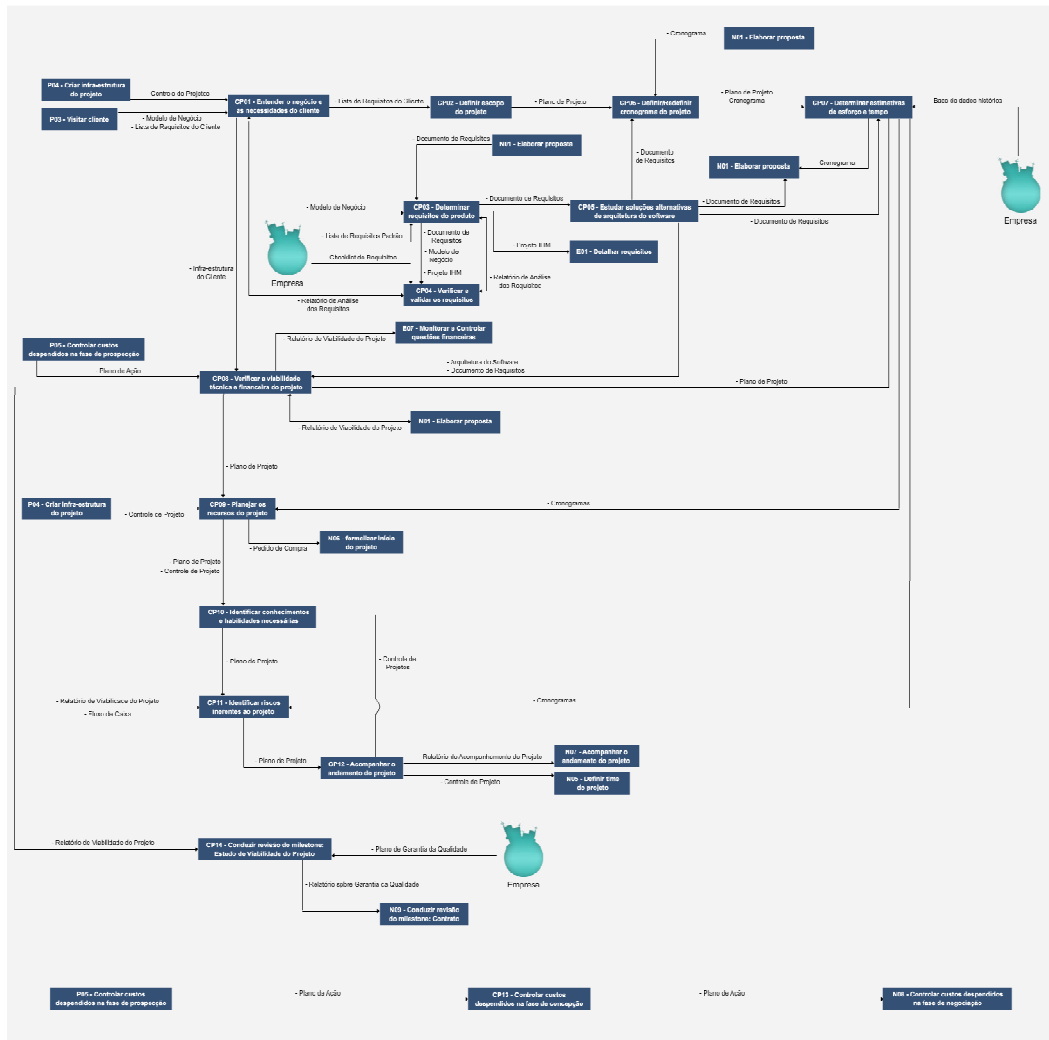


FIGURA A3.21 – VISÃO GRÁFICA – FASE: CONCEPÇÃO

QUADRO A3.21.1 – DETALHAMENTO DA ATIVIDADE CP01

Atividade	Cp01. Entender o negócio e as necessidades do cliente
Responsáveis	Gerente de Projeto Engenheiro de Requisitos Cliente
Descrição	Essa atividade tem por objetivo entender o negócio do cliente, as necessidades do cliente e as regras de seu negócio, por meio de técnicas de levantamentos de requisitos, a fim de definir com clareza os requisitos do sistema.
Recursos	- Ferramenta de modelagem de negócio (ex: Rational Rose, ARPO) - Técnica JAD - Ferramenta de Prototipação
Entradas	e1) Controle de Projetos e2) Modelo de Negócio e3) Lista de Requisitos do Cliente e4) Relatório de Análise dos Requisitos
Saídas	s1) Modelo de Negócio s2) Lista de Requisitos do Cliente s3) Infra-Estrutura do Cliente s4) Relatório de Análise dos Requisitos

QUADRO A3.21.1 – DETALHAMENTO DA ATIVIDADE CP01 (CONT.)

Tarefas	<p>No caso de já ter sido realizada a atividade "Cp04. Verificar e validar os requisitos" e, se necessário:</p> <ul style="list-style-type: none"> - Realizar reuniões com o cliente para elucidar os conflitos entre os requisitos detectados na peer review <ul style="list-style-type: none"> * Fazer as alterações no "Modelo de Negócio", caso necessário * Descrever a solução adotada no item 1 do "Relatório de Análise dos Requisitos" <p>Em outros casos:</p> <ul style="list-style-type: none"> - Identificar fontes de requisitos (documentos, pessoas, sistemas de informação utilizados). Por exemplo, caso haja um projeto que irá influenciar na maneira de trabalhar de toda a organização, é relevante saber quais são as pessoas chave de cada área da empresa, quais sistemas são utilizados em cada área, etc. - Selecionar técnicas de levantamento de requisitos. Por exemplo, entrevistas, questionários, brainstorming, prototipação, JAD, entre outras. Caso seja utilizada a técnica de prototipação, os protótipos podem ser registrados no Apêndice 3 do arquivo "Documento de Requisitos". - Preparar as reuniões de levantamento de requisitos, de acordo com as técnicas selecionadas. <ul style="list-style-type: none"> * Com base nas fontes de requisitos, definir os participantes das reuniões e as disponibilidades de cada um. * Agendar cronograma das reuniões * Preparar o material que será utilizado, de acordo com as técnicas selecionadas. Por exemplo, preparar um questionário, etc - Levantar/complementar as necessidades do cliente, a partir das reuniões, e compilar os dados obtidos, atualizando a "Lista de Requisitos do Cliente" (Apêndice 1 do Documento de Requisitos). - Entender o negócio do cliente <ul style="list-style-type: none"> * Caso ainda não exista, criar "Modelo de Negócio" (conforme GCf01) * Levantar/complementar os casos de uso de negócio e registrá-los no item 2 do arquivo "Modelo de Negócio". * Levantar/complementar as regras do negócio e vocabulário e registrá-los nos respectivos itens do arquivo "Modelo de Negócio". - Criar arquivo "Infra-Estrutura do Cliente" (conforme GCf01) e registrar sobre a infra-estrutura atual do cliente, pois influenciará na solução a ser proposta. - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"
----------------	--

QUADRO A3.21.2 – DETALHAMENTO DA ATIVIDADE CP02

Atividade	Cp02. Definir escopo do projeto
Responsáveis	Gerente de Projeto Engenheiro de Requisitos
Descrição	Essa atividade deverá ser realizada para dar início ao planejamento do projeto e para dar suporte à realização da proposta que deverá ser enviada ao cliente.
Recursos	
Entradas	e1) Lista de Requisitos do Cliente
Saídas	s1) Plano de Projeto
Tarefas	<ul style="list-style-type: none"> - Criar Plano do Projeto (conforme GCf01) - A partir do levantamento de requisitos, descrever o escopo do projeto do arquivo "Plano de Projeto". <ul style="list-style-type: none"> * Descrever o cenário (item 1.1): <ul style="list-style-type: none"> - identificar as necessidades que motivaram o cliente a solicitar o projeto, a partir da "Lista de Requisitos do Cliente" - definir os stakeholders (interessados) e usuários do sistema (quem são e quais as necessidades de cada um) e - descrever as expectativas que o projeto visa atender * Definir premissas, limitações e restrições do projeto (item 1.3) - Definir os critérios de aceitação do cliente e registrá-los no item 1.4 do "Plano de Projeto" <ul style="list-style-type: none"> * Descrever como o projeto atenderá às expectativas do cliente * Descrever itens de sucesso para o cliente, como antecipação de prazos

QUADRO A3.21.3 – DETALHAMENTO DA ATIVIDADE Cp03

Atividade	Cp03. Determinar requisitos do produto
Responsáveis	Engenheiro de Requisitos
Descrição	Transformar os requisitos do cliente em requisitos do produto, uma vez que uma necessidade do cliente pode ser traduzida em mais de um requisito do produto; a descrição dos requisitos dependerá do nível de detalhamento definido. Também, quando necessário, devem ser utilizados protótipos para auxiliar na melhor definição dos requisitos.
Recursos	
Entradas	e1) Modelo de Negócio e2) Documento de Requisitos e3) Lista de Requisitos Padrão do Produto (já existente na empresa) e4) Relatório de Análise dos Requisitos
Saídas	s1) Documento de Requisitos s2) Projeto IHM s3) Relatório de Análise dos Requisitos
Tarefas	<p>No caso de já ter sido realizada a atividade "Cp04. Verificar e validar os requisitos" e, se necessário:</p> <ul style="list-style-type: none"> - Realizar reuniões com o cliente para elucidar os conflitos entre os requisitos detectados na peer review <ul style="list-style-type: none"> * Fazer as alterações no "Documento de Requisitos", caso necessário * Descrever a solução adotada no item 2 do "Relatório de Análise dos Requisitos" <p>Em outros casos:</p> <ul style="list-style-type: none"> - Descrever/Atualizar o escopo do produto; as siglas, abreviações e definições utilizadas e as referências adquiridas e registrá-los no arquivo "Documento de Requisitos". <ul style="list-style-type: none"> * Criar/Atualizar o arquivo "Documento de Requisitos" (conforme GCf01) - Definir/Atualizar e registrar no capítulo 2 do "Documento de Requisitos", a partir da Lista de Requisitos do Cliente (apêndice do arquivo "Documento de Requisitos") e do "Modelo de Negócio": a Perspectiva do Produto, as principais Funções do Produto, as Características dos Usuários, os Limites, Suposições e Dependências <ul style="list-style-type: none"> * Caso se trate de algum produto já desenvolvido pela empresa, deve-se selecionar a partir da "Lista de Requisitos do Padrão Produto", aqueles requisitos que serão colocados na proposta ao cliente. - Definir/Atualizar volatilidade dos requisitos, isto é, probabilidade de mudança dos requisitos elicitados - Caso o sistema tenha que atender diferentes tipos de perfis de usuários: <ul style="list-style-type: none"> * Criar/Atualizar Projeto IHM (conforme GCf01) * Definir perfis de usuários e registrar no item 1 do "Projeto IHM" (Projeto de Interface Homem Máquina) - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

QUADRO A3.21.4 – DETALHAMENTO DA ATIVIDADE CP04

Atividade	Cp04. Verificar e Validar os requisitos
Responsáveis	Equipe de Teste Cliente Engenheiro de Requisitos
Descrição	Essa atividade tem por objetivo a realização da verificação dos requisitos e, posteriormente, da validação dos mesmos junto ao cliente.
Recursos	
Entradas	e1) Documento de Requisitos e2) Checklist de Requisitos e3) Modelo de Negócio e4) Projeto IHM e5) Relatório de Análise dos Requisitos
Saídas	s2) Relatório de Análise dos Requisitos
Tarefas	<p>- Realizar peer review (uma das pessoas não deve ter participado do desenvolvimento do Documento de requisitos)</p> <p>* Analisar os requisitos a partir do "Documento de Requisitos", "Modelo de Negócio" e "Projeto IHM" e, se for o caso, do "Relatório de Análise dos Requisitos" para ver se há conflitos entre as solicitações dos stakeholders, documentando tais conflitos no "Relatório de Análise dos Requisitos" (criar/atualizar "Relatório de Análise dos Requisitos", conforme GCf01)</p> <p>* Para dar subsídio ao peer review, pode ser utilizado um checklist (vide "Checklist de Requisitos")</p> <p>- Realizar reuniões com o cliente para validar os requisitos (averiguar se o que está descrito está de acordo com as necessidades do cliente)</p> <p>* Registrar eventuais conflitos no "Relatório de Análise dos Requisitos"</p>

QUADRO A3.21.5 – DETALHAMENTO DA ATIVIDADE CP05

Atividade	Cp05. Estudar soluções alternativas de arquitetura do software
Responsáveis	Equipe de Desenvolvimento Engenheiro de Requisitos
Descrição	Essa atividade é realizada com base nos requisitos mais críticos para ser sugerida a arquitetura do software.
Recursos	Ferramenta de Modelagem de Software (ex: ArgoUML, Rational Rose, Eclipse, JUDE)
Entradas	e1) Documento de Requisitos
Saídas	s1) Arquitetura do Software s2) Documento de Requisitos
Tarefas	<ul style="list-style-type: none"> - Criar/Atualizar o arquivo "Arquitetura do Software" (conforme GCf01) - Definir os Fatores Arquiteturais, ou seja, os requisitos não funcionais que influenciarão na escolha da arquitetura, e registrá-los no item 1 do documento "Arquitetura do Software". - Definir/Atualizar a arquitetura lógica do sistema, ou seja, a organização em larga escala das classes de software em pacotes, subsistemas, e dependendo do caso em camadas, e registrá-la no item 2 do documento "Arquitetura do Software". - Definir/Atualizar a topologia do sistema, ou seja, como os elementos físicos do sistema estarão dispostos e registrar no item 3 do documento "Arquitetura do Software". - Descrever as decisões arquiteturais e registrar no item 4 do documento "Arquitetura do Software". - Averiguar quais componentes podem ser adquiridos sem custos, quais já estão prontos e quais compensam ser comprados/terceirizados, e registrar no Apêndice 2 do "Documento de Requisitos". - Complementar o item "Perspectiva do Produto" do "Documento de Requisitos", dependendo do que foi definido como sendo a arquitetura do software. - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

QUADRO A3.21.6 – DETALHAMENTO DA ATIVIDADE CP06

Atividade	Cp06. Definir/Redefinir cronograma do projeto
Responsáveis	Gerente de Projeto
Descrição	Essa atividade tem como objetivo definir/redefinir o cronograma do projeto, a partir da WBS elaborada.
Recursos	Ferramenta de gerenciamento de projeto (ex: dot Project, MS-Project, Primavera)
Entradas	e1) Documento de Requisitos e2) Cronograma e3) Plano de Projeto
Saídas	s1) Cronograma s2) Plano de Projeto
Tarefas	<p>- Elaborar/Atualizar WBS (Work Breakdown Structure) no arquivo "Cronograma"</p> <p>* Pode ser usada uma ferramenta de gerenciamento de projeto, como dot Project</p> <p>* Definir/Redefinir quantos ciclos de desenvolvimento o projeto terá (cada ciclo de desenvolvimento passa, normalmente, pelas fases elaboração, construção e transição, com menor ou maior ênfase em algumas delas. Entretanto, pode existir a ocasião em que seja necessário voltar à fase de concepção para renegociar mudanças relevantes no projeto e, por consequência, no contrato). Cada ciclo de desenvolvimento produz um release do software.</p> <p>* Definir/Redefinir as fases do projeto, se achar pertinente</p> <p>* Definir/Redefinir as entregas do projeto, que são desdobramentos do produto do projeto em resultados tangíveis, a partir do "Documento de Requisitos". A característica fundamental das entregas é a capacidade de serem medidas e avaliadas. Por ex: manual do usuário, manual do sistema, versão do sistema.</p> <p>* Criar/Atualizar arquivo "Plano de Projeto" (conforme GCf01) e registrar quem são as entregas no escopo do projeto desse arquivo</p> <p>* Definir/Redefinir pacotes de trabalho os quais representam um conjunto de atividades que precisam ser feitas para obtenção de uma entrega ou de um produto de projeto.</p> <p>- Definir/Redefinir a inter-dependência e seqüência das atividades</p> <p>* Baseado nos pacotes de trabalho, definidos anteriormente, relacionar entradas e saídas de cada atividade</p> <p>* Definir/Redefinir quais as entradas e saídas são mutuamente dependentes</p> <p>- Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"</p>

QUADRO A3.21.7 – DETALHAMENTO DA ATIVIDADE Cp07

Atividade	Cp07. Determinar estimativas de esforço e tempo
Responsáveis	Gerente de Projeto Equipe de Desenvolvimento Engenheiro de Requisitos
Descrição	Essa atividade deverá ser realizada para estimar o esforço e duração das atividades do projeto com base em cronogramas de referência, base de dados histórica da empresa ou em experiências adquiridas, para que o projeto possa ser medido e controlado.
Recursos	Ferramenta de gerenciamento de projeto (ex: dot Project, MS-Project, Primavera)
Entradas	e1) Documento de Requisitos e2) Base de Dados Histórica da empresa e3) Cronograma e4) Plano de Projeto
Saídas	s1) Cronograma s2) Plano de Projeto
Tarefas	<ul style="list-style-type: none"> - Definir/Redefinir a dimensão (tamanho) do sistema a ser desenvolvido. <ul style="list-style-type: none"> * Podem ser usadas as métricas pontos por função, quantidade de casos de uso ou qualquer outra métrica que achar pertinente - Procurar algum projeto similar na Base de Dados Histórica da empresa (criada com o decorrer dos projetos da empresa), de acordo com alguns parâmetros, tais como: domínio de negócio do sistema, tamanho do sistema, funcionalidades do sistema, plataforma utilizada, nível de conhecimento dos colaboradores. - Calcular estimativas de esforço e duração em horas das atividades da WBS. <ul style="list-style-type: none"> * Criar/Atualizar arquivo "Cronograma" (conforme GCf01) * Os cálculos devem ser registrados no "Cronograma" * Utilizar a Base de Dados Histórica da empresa ou experiência de especialistas, além do "Documento de Requisitos" do projeto em andamento para embasar as estimativas de esforço e duração das atividades a serem realizadas. - Definir/Redefinir os papéis que devem executar cada uma das atividades e registrar no "Cronograma". Por exemplo, a atividade elaborar cronograma deve ser executada por um gerente de projeto e não por um colaborador da equipe de teste - Colocar no cronograma o valor da hora do papel que deve executar cada atividade, a fim de calcular o custo de cada atividade, e do projeto como um todo - Estabelecer/Restabelecer prazos máximos no escopo do projeto do "Plano de Projeto", a partir do cronograma elaborado. - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

QUADRO A3.21.8 – DETALHAMENTO DA ATIVIDADE CP08

Atividade	Cp08. Verificar a viabilidade técnica e financeira do projeto
Responsáveis	Gerente de Projeto Equipe de Vendas
Descrição	Essa atividade tem como objetivo avaliar se é viável a execução do projeto em termos técnicos e financeiros.
Recursos	Ferramenta de gerenciamento de projeto (ex: dot Project, MS-Project, Primavera)
Entradas	e1) Documento de Requisitos e2) Plano de Projeto e3) Arquitetura do Software e4) Plano de Ação e5) Infra-Estrutura do Cliente
Saídas	s1) Plano de Projeto s2) Relatório de Viabilidade do Projeto s3) Fluxo de Caixa
Tarefas	<ul style="list-style-type: none"> - Definir/Redefinir o custo operacional geral do produto para a empresa desenvolvedora <ul style="list-style-type: none"> * Criar/Atualizar arquivo "Relatório de Viabilidade" (conforme GCf01) * Descrever os itens da configuração técnica para a solução a ser concebida, a partir do documento "Arquitetura do Software" e registrar na "Tabela de Custos" do "Relatório de Viabilidade". * Descrever os outros direcionadores de custo (cost driver), tal como valor da mão-de-obra * Definir/Redefinir valor da unidade dos cost drivers * Estimar valores de possíveis aquisições * Considerar as restrições do projeto descritas no escopo do projeto no arquivo "Plano de Projeto" - Descrever relação custo x benefício para o cliente da solução proposta e registrar no "Relatório de Viabilidade", considerando as informações contidas na "Infra-Estrutura do Cliente" - Fazer/Refazer o fluxo de caixa do projeto <ul style="list-style-type: none"> * Criar/Atualizar arquivo "Fluxo de Caixa" (conforme GCf01) * Utilizar dados relacionados aos custos até o fechamento do contrato, advindos do "Plano de Ação" - Definir/Redefinir o cronograma de recebimentos e registrar no "Relatório de Viabilidade" - Definir/Redefinir o preço do produto final do projeto e registrar no "Relatório de Viabilidade" - Registrar custo operacional geral e preço-meta do produto do projeto no escopo do projeto do "Plano de Projeto" - Verificar no Fluxo de Caixa, o valor que poderá ser investido no projeto por mês, para que possam ser alocados colaboradores e recursos de forma otimizada e que não inviabilize a questão financeira do projeto para a empresa desenvolvedora - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

QUADRO A3.21.9 – DETALHAMENTO DA ATIVIDADE CP09

Atividade	Cp09. Planejar os recursos do projeto
Responsáveis	Gerente de Projeto Equipe de Compras
Descrição	Essa atividade tem como objetivo levantar quais são os recursos necessários para o desenvolvimento do projeto. Entende-se por recursos todos os equipamentos, ferramentas e materiais.
Recursos	Ferramenta de gerenciamento de projeto (ex: dot Project, MS-Project, Primavera)
Entradas	e1) Cronograma e2) Controle de Projetos e3) Plano de Projeto
Saídas	s1) Plano de Projeto s2) Controle de Projetos s3) Pedido de Compra
Tarefas	<p>- Verificar no "Cronograma" quais atividades ou entregas necessitam de recursos e preencher no item "Plano de Recursos" do "Plano de Projeto"</p> <p>- Verificar na planilha "Identificação de Recursos" do arquivo "Controle de Projetos" se os recursos necessários existem e estão disponíveis ou se precisam ser adquiridos.</p> <p>* Se existirem e estiverem disponíveis, o gerente de projetos deve assinalar na planilha "Alocação de Recursos" do "Controle de Projetos" a data de alocação e a previsão de uso desse recurso em qual projeto.</p> <p>* Se não existirem, criar o arquivo "Pedido de Compra" (conforme CGf01) e preenchê-lo. Inserir no "Pedido de Compra" o status "congelado", que será alterado para "liberado" caso o cliente feche o contrato.</p>

QUADRO A3.21.10 – DETALHAMENTO DA ATIVIDADE CP10

Atividade	Cp10. Identificar conhecimentos e habilidades necessárias
Responsáveis	Gerente de Projeto
Descrição	Essa atividade tem como objetivo levantar quais são os conhecimentos e habilidades necessárias para o desenvolvimento do projeto.
Recursos	Ferramenta de gerenciamento de projeto (ex: dot Project, MS-Project, Primavera)
Entradas	e1) Controle de Projetos e2) Plano de Projeto
Saídas	s1) Controle de Projetos s2) Plano de Projeto
Tarefas	<ul style="list-style-type: none"> - Identificar quais habilidades/conhecimentos serão necessários para o desenvolvimento do projeto e registrar no item "Plano de Habilidades e Conhecimento" do "Plano de Projeto" - Verificar no item "Plano de Recursos" do "Plano de Projeto", quais ferramentas serão utilizadas no projeto para confrontar na planilha "identificação de colaboradores" do arquivo "Controle de Projetos" quais são os mais indicados para fazer parte do time - Sugerir o time, baseado na análise realizada anteriormente, e registrá-lo junto ao "Plano de Habilidades e Conhecimento" do "Plano de Projeto" - Registrar na planilha "Alocação de Colaboradores", os colaboradores do projeto que foram sugeridos para compor o time, a possível data de alocação, previsão do término da alocação - Definir status dos colaboradores pertencentes ao time como "reservado" na planilha "Alocação de Colaboradores" do "Controle de Projetos" - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

QUADRO A3.21.11 – DETALHAMENTO DA ATIVIDADE CP11

Atividade	Cp11. Identificar riscos inerentes ao projeto
Responsáveis	Gerente de Projeto Analista de Riscos
Descrição	Essa atividade tem como objetivo identificar os riscos inerentes ao projeto e estabelecer um plano de contingência.
Recursos	
Entradas	e1) Cronograma e2) Relatório de Viabilidade do Projeto e3) Fluxo de Caixa e4) Plano de Projeto
Saídas	s1) Plano de Projeto
Tarefas	<ul style="list-style-type: none"> - Identificar os riscos orçamentários analisando o relatório de viabilidade - Identificar os riscos de prazo verificando o caminho crítico (PERT/CPM) no cronograma - Identificar os riscos de rotatividade de pessoal baseado no histórico da empresa - Identificar riscos técnicos, riscos de projeto e de negócio - Definir a escala dos riscos: booleana, qualitativa ou quantitativa. Por ex: booleana: sim ou não; qualitativa: altamente, pouco, moderado, etc; quantitativo: em termos numéricos como % - Elaborar/Atualizar checklists (conjuntos de questões) para classificar os riscos ou utilizar checklists existentes, baseado na escala definida anteriormente - Definir/Redefinir a probabilidade de ocorrência dos riscos baseado na classificação anterior - Definir/Redefinir os impactos dos riscos, ou seja, quais as conseqüências caso os riscos ocorram - Definir/Redefinir prioridade dos riscos a serem "atacados", uma vez que se conhece os impactos dos mesmos - Definir/Redefinir ações pró-ativas para não deixar que os riscos negativos priorizados se concretizem e para garantir que os riscos positivos priorizados se concretizem - Elaborar/Atualizar o plano de contingência que será utilizado caso algum risco venha a se concretizar e registrá-lo no apêndice "Plano de Contingência" do "Plano de Projeto" - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

QUADRO A3.21.12 – DETALHAMENTO DA ATIVIDADE CP12

Atividade	Cp12. Acompanhar o andamento do projeto
Responsáveis	Gerente de Projeto
Descrição	Essa atividade tem como objetivo o acompanhamento do projeto.
Recursos	
Entradas	e1) Controle de Projetos e2) Plano de Projeto
Saídas	s1) Controle de Projetos s2) Relatório de Acompanhamento do Projeto
Tarefas	<ul style="list-style-type: none"> - Inserir a data de término da fase do projeto na planilha "Acompanhamento de Projetos" do arquivo "Controle de Projetos", replicar a linha do projeto e inserir a nova fase com sua data de início - Inserir na planilha "Acompanhamento de Projetos" do arquivo "Controle de Projetos" a quantidade de horas trabalhadas de cada um dos colaboradores nessa fase. - Revisar a documentação quanto aos riscos no arquivo "Plano de Projeto" no contexto atual do projeto <ul style="list-style-type: none"> * Em casos de problemas, acionar o "Plano de Contingência" (Apêndice 1 do "Plano de Projeto") * Criar arquivo "Relatório de Acompanhamento do Projeto" (conforme GCf01) * Registrar informações no arquivo "Relatório de Acompanhamento do Projeto" referentes ao acionamento do "Plano de Contingência"

QUADRO A3.21.13 – DETALHAMENTO DA ATIVIDADE CP13

Atividade	Cp13. Controlar custos despendidos na fase de concepção
Responsáveis	Equipe de Vendas Gerente de Projeto
Descrição	Essa atividade tem como objetivo registrar todos os custos referentes à execução das atividades dessa fase.
Recursos	
Entradas	e1) Plano de Ação
Saídas	s1) Plano de Ação
Tarefas	<ul style="list-style-type: none"> - Atualizar na planilha "Acompanhamento" (item "Atividades") do arquivo "Plano de Ação" a quantidade de atividades realizadas de fato. - Atualizar na planilha "Acompanhamento" (item "Custos") os custos reais referentes à execução de todas as atividades dessa fase.

QUADRO A3.21.14 – DETALHAMENTO DA ATIVIDADE CP14

Atividade	Cp14. Conduzir revisão do milestone: Estudo de Viabilidade do Projeto
Responsáveis	Equipe de Qualidade Gerente de Projeto
Descrição	Essa atividade tem o objetivo de revisar o milestone que marca o final dessa fase.
Recursos	
Entradas	e1) Plano de Garantia da Qualidade (já existente na empresa) e2) Produto de Trabalho: Relatório de Viabilidade do Projeto
Saídas	s1) Relatório sobre Garantia da Qualidade
Tarefas	- Realizar a atividade "GqPP01. Avaliar o processo e assegurar resolução de não conformidades" - Realizar a atividade "GqPP02. Avaliar os produtos de trabalho e assegurar resolução de não conformidades" - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

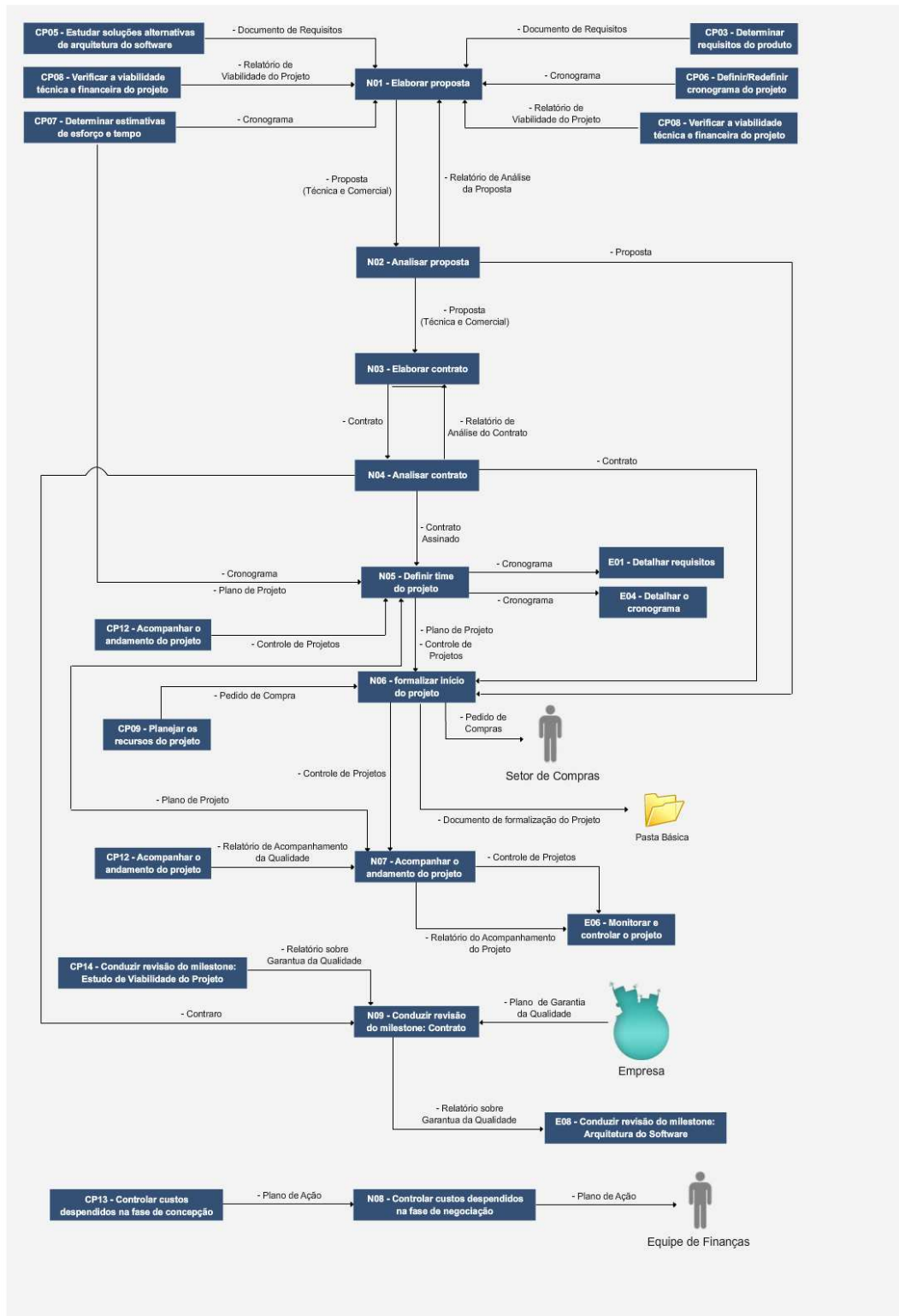


FIGURA A3.22 – VISÃO GRÁFICA – FASE: NEGOCIAÇÃO

QUADRO A3.22.1 – DETALHAMENTO DA ATIVIDADE N01

Atividade	N01. Elaborar proposta
Responsáveis	Gerente do Projeto Equipe de Vendas Engenheiro de Requisitos
Descrição	Elaborar a proposta do projeto baseado no Documento de Requisitos e no cronograma de recebimentos estipulado no Relatório de Viabilidade do Projeto, além de considerar o feedback do cliente encontrado no Relatório de Análise da Proposta.
Recursos	
Entradas	e1) Documento de Requisitos e2) Relatório de Viabilidade do Projeto e3) Relatório de Análise da Proposta e4) Cronograma
Saídas	s1) Proposta Técnica s2) Proposta Comercial s3) Relatório de Viabilidade do Projeto s4) Cronograma s5) Documento de Requisitos
Tarefas	<ul style="list-style-type: none"> - Criar/Atualizar arquivo "Proposta Comercial" (conforme GCf01) e preencher: <ul style="list-style-type: none"> * os dados completos do cliente (Razão Social, CNPJ, Endereço, Telefones, Contato) * a relação custo x benefício da solução proposta para o cliente, advinda do "Relatório de Viabilidade do Projeto" * o cronograma estimado de entregas baseado no Cronograma (WBS) * o preço baseado no "Relatório de Viabilidade do Projeto" * o cronograma de pagamentos do cliente baseado no cronograma de recebimentos do "Relatório de Viabilidade do Projeto" - Determinar a validade da "Proposta Comercial" - Criar/Atualizar arquivo "Proposta Técnica" (conforme GCf01) e: <ul style="list-style-type: none"> * registrar os dados da empresa de desenvolvimento * inserir, como apêndice, na "Proposta Técnica" os requisitos descritos no item "2.2 Funções do Produto" do arquivo "Documento de Requisitos" - Registrar o ID da "Proposta Técnica" na "Proposta Comercial" - Analisar o "Relatório de Análise da Proposta" vindo do cliente com o "Relatório de Viabilidade do Projeto" - Realizar as possíveis modificações na proposta e, se necessário: <ul style="list-style-type: none"> * acessar a atividade "Cp03. Determinar requisitos do produto" para modificar o "Documento de Requisitos" * acessar a atividade "Cp06. Definir/Redefinir cronograma do projeto" para modificar o cronograma estimado de entregas do "Cronograma" (WBS) * acessar a atividade "Cp08. Verificar a viabilidade técnica e financeira do projeto" para modificar o cronograma de recebimentos do "Relatório de Viabilidade do Projeto" - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

QUADRO A3.22.2 – DETALHAMENTO DA ATIVIDADE N02

Atividade	N02. Analisar proposta
Responsáveis	Cliente
Descrição	O cliente deve analisar a proposta; no caso de estar de acordo deve devolvê-la assinada, caso contrário, deve enviar as alterações necessárias que serão registradas como um Relatório de Análise da Proposta.
Recursos	
Entradas	e1) Proposta (Comercial e Técnica)
Saídas	s2) Relatório de Análise da Proposta
Tarefas	- Obter o feedback do cliente a respeito da proposta enviada. * Caso o cliente esteja de acordo, deve assinar a proposta

QUADRO A3.22.3 – DETALHAMENTO DA ATIVIDADE N03

Atividade	N03. Elaborar contrato
Responsáveis	Equipe de Vendas Gerente de projetos Assessor Jurídico
Descrição	Elaborar o contrato do projeto baseado na Proposta assinada pelo cliente, além de considerar o feedback dado pelo cliente encontrado no Relatório de Análise do Contrato.
Recursos	
Entradas	e1) Proposta e2) Relatório de Análise do Contrato
Saídas	s1) Contrato
Tarefas	- Criar/Atualizar arquivo "Contrato" (conforme GCf01) e preencher: * os dados do cliente (obtidos da proposta comercial) * os dados da empresa de desenvolvimento (obtidos da proposta técnica) * o objeto de contrato * os aspectos técnicos: prazo de desenvolvimento e data das entregas, especificação quanto ao suporte, especificação quanto às condições de instalação e uso do sistema * os aspectos financeiros: preço, cronograma de pagamentos e penalidades * os aspectos legais: estabelecimento de garantias, definições de prazo do contrato e de licenças de uso, definição das responsabilidades das partes * o controle de alterações: procedimentos de mudanças e limitação para alteração de requisitos - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

QUADRO A3.22.4 – DETALHAMENTO DA ATIVIDADE N04

Atividade	N04. Analisar contrato
Responsáveis	Cliente
Descrição	O cliente deve analisar o contrato; no caso de estar de acordo deve devolvê-lo assinado, caso contrário, deve enviar as alterações necessárias que serão registradas como um Relatório de Análise do Contrato.
Recursos	
Entradas	e1) Contrato
Saídas	s1) Relatório de Análise do Contrato
Tarefas	- Obter o feedback do cliente a respeito do contrato enviado, referente às questões não tratadas na proposta, como questões legais e controle de alterações * Caso o cliente esteja de acordo, deve assinar o contrato

QUADRO A3.22.5 – DETALHAMENTO DA ATIVIDADE N05

Atividade	N05. Definir time do projeto
Responsáveis	Gerente de Projeto
Descrição	Definir quem serão as pessoas integrantes do time do projeto com base no Plano de Habilidades e Conhecimento (do Plano de Projeto), no Cronograma estabelecido e no calendário de colaboradores (obtido pela análise da planilha "alocação de colaboradores" do Controle de Projetos).
Recursos	Ferramenta de gerenciamento de projeto (ex: dot Project, MS-Project, Primavera)
Entradas	e1) Cronograma e2) Plano de Projeto e3) Controle de Projetos e4) Contrato assinado
Saídas	s1) Plano de Projeto s2) Cronograma s3) Controle de Projetos
Tarefas	- Verificar no Calendário de Alocação de Colaboradores (obtido pela análise da planilha "alocação de colaboradores" do Controle de Projetos) a disponibilidade dos colaboradores, confrontando com o time sugerido no Plano de Habilidades e Conhecimento do "Plano de Projeto" * Reajustar o seqüenciamento das atividades ou alocar algumas das atividades para outro colaborador, caso haja acúmulo de atividades para um colaborador * Definir o time do projeto e atualizar no Plano de Habilidades e Conhecimento do "Plano de Projeto" e na planilha de alocação de colaboradores do "Controle de Projetos", inserindo o status "formalizado" para os colaboradores escolhidos - Atualizar o Cronograma substituindo os papéis pelos colaboradores do time de projeto - Definir a estratégia utilizada durante o desenvolvimento do projeto, tal como reuniões semanais com o cliente, e registrar no escopo do projeto do "Plano de Projeto"

QUADRO A3.22.6 – DETALHAMENTO DA ATIVIDADE N06

Atividade	N06. Formalizar início do projeto
Responsáveis	Gerente de Projeto Equipe de Vendas Equipe de Qualidade Equipe de Finanças Equipe de Compras
Descrição	Comunicar a todos os interessados, interna e externamente ao projeto, a inicialização do mesmo.
Recursos	
Entradas	e1) Contrato e2) Proposta Técnica e3) Pedido de Compra e4) Controle de Projetos e5) Plano de Projeto
Saídas	s1) Documento de formalização do projeto s2) Pedido de Compra s3) Controle de Projetos
Tarefas	<ul style="list-style-type: none"> - Comunicar internamente a formalização do projeto, uma vez que foi iniciado na fase de Prospecção, disponibilizando aos colaboradores a Proposta Técnica - Alterar o status do projeto (na planilha "Acompanhamento de Projetos" do arquivo "Controle de Projetos") para "formalizado" e preencher a data de formalização - Emitir a Nota Fiscal e registrar a emissão na aba "Notas Fiscais" do arquivo "Controle de Projetos" - Lançar as duplicatas e registrar o lançamento na aba "Notas Fiscais" do arquivo "Controle de Projetos" - Comunicar ao setor de faturamento os registros do contrato - Alterar o status do "Pedido de Compra" para "liberado" com a data de liberação, se for o caso, e acessar as atividades "Acq01. Selecionar fornecedores e estabelecer acordo" e "Acq02. Aceitar o produto adquirido" - Obter comprometimento com o plano <ul style="list-style-type: none"> * Criar arquivo "Documento de Formalização do Projeto" (conforme GCf01) * Assegurar comprometimento dos colaboradores com o "Plano de Projeto" por meio de uma reunião que deve ser documentada em forma de ata no "Documento de Formalização do Projeto", assinado por todos os colaboradores e anexado na pasta física do projeto.

QUADRO A3.22.7 – DETALHAMENTO DA ATIVIDADE N07

Atividade	N07. Acompanhar o andamento do projeto
Responsáveis	Gerente de Projeto
Descrição	Essa atividade tem como objetivo o acompanhamento do projeto.
Recursos	
Entradas	e1) Controle de Projetos e2) Plano de Projeto e3) Relatório de Acompanhamento do Projeto
Saídas	s1) Controle de Projetos s2) Relatório de Acompanhamento do Projeto
Tarefas	<ul style="list-style-type: none"> - Inserir a data de término da fase do projeto na planilha "Acompanhamento de Projetos" do arquivo "Controle de Projetos", replicar a linha do projeto e inserir a nova fase com sua data de início - Inserir na planilha "Acompanhamento de Projetos" do arquivo "Controle de Projetos" a quantidade de horas trabalhadas de cada um dos colaboradores nessa fase. - Revisar a documentação quanto aos riscos no arquivo "Plano de Projeto" no contexto atual do projeto <ul style="list-style-type: none"> * Em casos de problemas, acionar o "Plano de Contingência" (Apêndice 1 do "Plano de Projeto") * Criar/Atualizar arquivo "Relatório de Acompanhamento do Projeto" (conforme GCf01) * Registrar informações no arquivo "Relatório de Acompanhamento do Projeto" referentes ao acionamento do "Plano de Contingência" - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

QUADRO A3.22.8 – DETALHAMENTO DA ATIVIDADE N08

Atividade	N08. Controlar custos despendidos na fase de negociação
Responsáveis	Equipe de Vendas Gerente de Projeto
Descrição	Essa atividade tem como objetivo registrar todos os custos referentes à execução das atividades dessa fase.
Recursos	
Entradas	e1) Plano de Ação
Saídas	s1) Plano de Ação
Tarefas	<ul style="list-style-type: none"> - Atualizar na planilha "Acompanhamento" (item "Atividades") do arquivo "Plano de Ação" a quantidade de atividades realizadas de fato. - Atualizar na planilha "Acompanhamento" (item "Custos") os custos reais referentes à execução de todas as atividades dessa fase.

QUADRO A3.22.9 – DETALHAMENTO DA ATIVIDADE N09

Atividade	N09. Conduzir revisão do milestone: Contrato
Responsáveis	Equipe de Qualidade Gerente de Projeto
Descrição	Essa atividade tem o objetivo de revisar o milestone que marca o final dessa fase.
Recursos	
Entradas	e1) Plano de Garantia da Qualidade e2) Produto de Trabalho: Contrato e3) Relatório sobre Garantia da Qualidade
Saídas	s1) Relatório sobre Garantia da Qualidade
Tarefas	- Realizar a atividade "GqPP01. Avaliar o processo e assegurar resolução de não conformidades" - Realizar a atividade "GqPP02. Avaliar os produtos de trabalho e assegurar resolução de não conformidades" - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

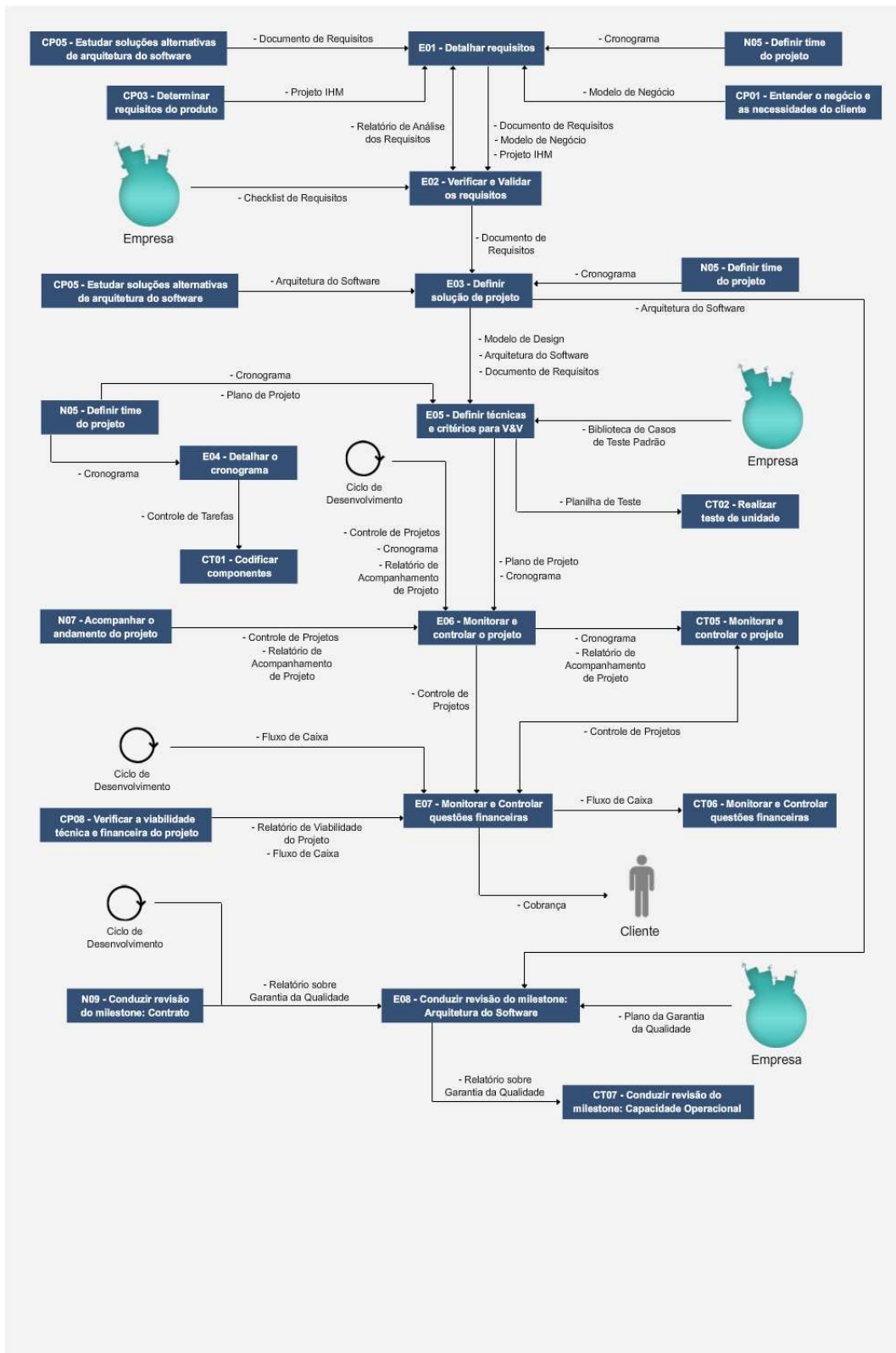


FIGURA A3.23 – VISÃO GRÁFICA – FASE: ELABORAÇÃO

QUADRO A3.23.1 – DETALHAMENTO DA ATIVIDADE E01

Atividade	E01. Detalhar requisitos
Responsáveis	Engenheiro de Requisitos Equipe de Desenvolvimento
Descrição	Essa atividade tem o objetivo de detalhar os requisitos já obtidos, ao ponto de ser mais útil aos programadores.
Recursos	- Ferramenta de Modelagem de Dados (ex: Dia, ArgoUML, ERWIN, Rational Rose) - Ferramenta de prototipação
Entradas	e1) Documento de Requisitos e2) Projeto IHM e3) Cronograma e4) Modelo de Negócio
Saídas	s1) Documento de Requisitos s2) Projeto IHM s3) Modelo de Negócio
Tarefas	<p>No caso de já ter sido realizada a atividade "E02. Verificar e validar os requisitos" e, se necessário:</p> <ul style="list-style-type: none"> - Realizar reuniões com o cliente para elucidar os conflitos entre os requisitos detectados na peer review <ul style="list-style-type: none"> * Fazer as alterações no "Documento de Requisitos", caso necessário * Fazer as alterações no "Modelo de Negócio", caso necessário * Fazer as alterações no "Projeto IHM", caso necessário * Descrever a solução adotada no item 2 do "Relatório de Análise dos Requisitos" <p>Em outros casos:</p> <ul style="list-style-type: none"> - Para cada incremento a ser criado no ciclo de desenvolvimento, definido no "Cronograma: <ul style="list-style-type: none"> * Se necessário, detalhar/atualizar o "Modelo de Negócio" * Detalhar/Atualizar os requisitos funcionais do item 2.2 do "Documento de Requisitos" e registrar no item 3.4 deste documento * Elaborar/Atualizar o Modelo Conceitual e registrá-lo no item 3.4 do "Documento de Requisitos" * Detalhar/Atualizar os requisitos de interface externa e outros requisitos não funcionais (tais como requisitos de documentação, de manutenção) do capítulo 2 do Documento de Requisitos e registrar no capítulo 3 deste documento * Elaborar/Atualizar o Projeto IHM <ul style="list-style-type: none"> - Caso o perfil dos usuários não tenha sido definido na fase de concepção, criar Projeto IHM (conforme GCf01), obter perfil dos usuários e registrar no "Projeto IHM" - A partir do perfil dos usuários, definir os princípios gerais do projeto (com base em critérios ergonômicos) <ul style="list-style-type: none"> - Definir/Atualizar o padrão de telas no arquivo "Projeto IHM" - Definir/Atualizar o mapa de navegação no arquivo "Projeto IHM" - Criar/Atualizar a matriz de rastreabilidade entre requisitos do cliente e do produto no Documento de Requisitos - Criar/Atualizar a matriz de rastreabilidade entre requisitos funcionais e não funcionais do produto no Documento de Requisitos - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

QUADRO A3.23.2 – DETALHAMENTO DA ATIVIDADE E02

Atividade	E02. Verificar e Validar os requisitos
Responsáveis	Equipe de Teste Cliente Engenheiro de Requisitos
Descrição	Essa atividade tem por objetivo a realização da verificação dos requisitos e, posteriormente, da validação dos mesmos junto ao cliente.
Recursos	
Entradas	e1) Documento de Requisitos e2) Checklist de Requisitos e3) Modelo de Negócio e4) Projeto IHM e5) Relatório de Análise dos Requisitos
Saídas	s1) Relatório de Análise dos Requisitos
Tarefas	<p>- Realizar peer review (uma das pessoas não deve ter participado do desenvolvimento do Documento de requisitos)</p> <p>* Analisar os requisitos a partir do "Documento de Requisitos", "Modelo de Negócio" e "Projeto IHM" e, se for o caso, do "Relatório de Análise dos Requisitos" para ver se há conflitos entre as solicitações dos stakeholders, documentando tais conflitos no "Relatório de Análise dos Requisitos" (criar/atualizar "Relatório de Análise dos Requisitos", conforme GCf01)</p> <p>* Para dar subsídio ao peer review, pode ser utilizado um checklist (vide "Checklist de Requisitos")</p> <p>- Realizar reuniões com o cliente para validar os requisitos (averiguar se o que está descrito está de acordo com as necessidades do cliente)</p> <p>* Registrar eventuais conflitos no "Relatório de Análise dos Requisitos"</p>

QUADRO A3.23.3 – DETALHAMENTO DA ATIVIDADE E03

Atividade	E03. Definir solução de projeto
Responsáveis	Equipe de Desenvolvimento Engenheiro de Requisitos
Descrição	Essa atividade tem o objetivo de elaborar uma solução de projeto com base na arquitetura de software estabelecida.
Recursos	- Ferramenta de Modelagem de Software (ex: ArgoUML, Dia, Rational Rose, JUDE) - Ferramenta de Modelagem de Dados (ex: ERWIN)
Entradas	e1) Arquitetura do Software e2) Documento de Requisitos e3) Cronograma
Saídas	s1) Arquitetura do Software s2) Documento de Requisitos s3) Modelo de Design
Tarefas	<ul style="list-style-type: none"> - Confirmar a arquitetura do software projetada na fase de concepção <ul style="list-style-type: none"> * Caso seja necessário, podem ser realizadas modificações na "Arquitetura do Software" - Para cada incremento a ser criado no ciclo de desenvolvimento, definido no "Cronograma" <ul style="list-style-type: none"> * Criar/Atualizar "Modelo de Design" (conforme GCf01) <ul style="list-style-type: none"> - Elaborar o Modelo Lógico do Sistema com base no Documento de Requisitos (T) - Criar o modelo físico de dados correspondente ao Modelo de Dados (parte do modelo lógico do sistema) - Criar/Atualizar a matriz de rastreabilidade entre requisitos do produto do Documento de Requisitos e componentes do Modelo de Design e registrar no "Documento de Requisitos" - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

QUADRO A3.23.4 – DETALHAMENTO DA ATIVIDADE E04

Atividade	E04. Detalhar o cronograma
Responsáveis	Equipe de Desenvolvimento Gerente de Projeto
Descrição	Essa atividade tem como objetivo o detalhamento do cronograma, por parte dos programadores. Assim, cada um deles será responsável por gerenciar suas tarefas, respeitando o prazo a ele estabelecido no cronograma.
Recursos	
Entradas	e1) Cronograma
Saídas	s1) Controle de tarefas
Tarefas	<ul style="list-style-type: none"> - Para cada incremento a ser criado no ciclo de desenvolvimento, definido no "Cronograma" <li style="padding-left: 20px;">* Definir/Atualizar as tarefas a serem realizadas para atender à(s) atividade(s) estabelecida(s) no "Cronograma" <li style="padding-left: 20px;">* Cada programador deve criar/atualizar o arquivo "Controle de Tarefas" (conforme GCf01), registrar suas tarefas em uma "Lista de Tarefas" e será responsável por controlar suas tarefas diariamente - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

QUADRO A3.23.5 – DETALHAMENTO DA ATIVIDADE E05

Atividade	E05. Definir técnicas e critérios para V&V
Responsáveis	Equipe de Desenvolvimento Equipe de Teste Gerente de Projeto
Descrição	Essa atividade tem o objetivo de definir quais as técnicas e critérios de verificação e validação serão utilizados para avaliar os componentes a serem construídos.
Recursos	
Entradas	e1) Plano de Projeto e2) Cronograma e3) Biblioteca de Casos de Teste Padrão e4) Documento de Requisitos e5) Modelo de Design e6) Arquitetura do Software
Saídas	s1) Planilha de Teste s2) Plano de Projeto s3) Cronograma

QUADRO A3.23.5 – DETALHAMENTO DA ATIVIDADE E05 (CONT.)

<p>Tarefas</p>	<ul style="list-style-type: none"> - Considerar o tempo estimado no Cronograma (WBS) para elaborar os casos de teste - Planejar/Atualizar Teste de Unidade <ul style="list-style-type: none"> * Definir os critérios de teste correspondentes às técnicas de teste a serem utilizados e registrar no item "Plano de Teste" do "Plano de Projeto". Por exemplo: critérios particionamento de classes em equivalência e análise do valor limite da categoria de técnicas funcionais; critérios Todos-Arcos da técnica estrutural; critério Análise de Mutantes da categoria de técnicas baseadas em erros. * Estabelecer os casos de teste a serem executados considerando cada um dos critérios definidos, com base na Arquitetura do Software, Modelo de Design e Documento de Requisitos. Para auxiliar nessa tarefa, pode-se recorrer à biblioteca de casos de teste padrão da empresa (caso exista). Criar/Atualizar arquivo "Planilha de Teste" (conforme GCf01) e registrar os casos de teste a serem executados na planilha "Casos de Teste de Unidade". - Planejar/Atualizar Teste de Integração <ul style="list-style-type: none"> * Definir o ambiente para a integração e teste. Esse ambiente pode incluir o reuso de recursos organizacionais existentes, mas pode ser necessária a aquisição de equipamentos (nesse caso, redirecionar para as atividades do processo de apoio "Aquisição") * Definir a estratégia de integração (pode ser: top-down ou botton-up), definir os critérios de teste a serem utilizados e e registrá-los no item "Plano de Teste" do "Plano de Projeto" * Estabelecer os casos de teste, de acordo com a Arquitetura do Software, Modelo de Design e Documento de Requisitos. Para auxiliar essa tarefa pode-se recorrer à biblioteca de casos de teste padrão da empresa (caso exista). Preencher na planilha "Casos de Teste de Integração" no arquivo "Planilha de Teste" a lista de casos de teste a serem executados. - Detalhar o "Cronograma" estabelecendo o tempo de execução dos casos de testes - Planejar/Atualizar Teste de Validação <ul style="list-style-type: none"> * Definir o tempo máximo para fazer o teste de validação, com base no Cronograma, e registrar no item "Plano de Teste" do "Plano de Projeto" * Estabelecer os casos de teste advindos da empresa desenvolvedora e do cliente, com base nos critérios de avaliação definidos no "Documento de Requisitos". Preencher na planilha "Casos de teste de validação" no arquivo "Planilha de Teste" a lista de casos de teste a serem executados. - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"
-----------------------	--

QUADRO A3.23.6 – DETALHAMENTO DA ATIVIDADE E06

Atividade	E06. Monitorar e controlar o projeto
Responsáveis	Gerente de Projeto
Descrição	Essa atividade tem como objetivo o monitoramento e controle do projeto.
Recursos	Ferramenta de gerenciamento de projeto (ex: dot Project, MS-Project, Primavera)
Entradas	e1) Plano de Projeto e2) Relatório de Acompanhamento de Projeto e3) Controle de Projetos e4) Cronograma
Saídas	s1) Relatório de Acompanhamento de Projeto s2) Controle de Projetos s3) Cronograma
Tarefas	<p>- Em qualquer iteração e ciclo de desenvolvimento:</p> <ul style="list-style-type: none"> * Inserir a data de término da fase do projeto na planilha "Acompanhamento de Projetos" do arquivo "Controle de Projetos", replicar a linha do projeto e inserir a nova fase com sua data de início * Inserir na planilha "Acompanhamento de Projetos" do arquivo "Controle de Projetos" a quantidade de horas trabalhadas de cada um dos colaboradores nessa fase. * Inserir no "Cronograma" a quantidade de horas trabalhadas de cada um dos colaboradores nessa fase. * Verificar no "Cronograma" o andamento das atividades executadas em relação às atividades planejadas <ul style="list-style-type: none"> - Identificar e gerenciar dependências em relação às datas planejadas - Identificar compromissos que não foram satisfeitos - Identificar a ocorrência de algum risco levantado no "Plano de Projeto" e, nesse caso, executar o "Plano de Contingência" documentado no "Plano de Projeto" - Criar/Atualizar o arquivo "Relatório de Acompanhamento de Projeto" (conforme GCf01) e registrar a revisão de andamento do projeto nesse relatório * Planejar e executar ações corretivas, caso necessário, e registrá-las no "Relatório de Acompanhamento de Projeto" <p>- Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"</p>

QUADRO A3.23.7 – DETALHAMENTO DA ATIVIDADE E07

Atividade	E07. Monitorar e Controlar questões financeiras
Responsáveis	Gerente de Projeto Equipe de Finanças
Descrição	Essa atividade tem como objetivo o monitoramento e controle de todas as questões financeiras referentes ao projeto.
Recursos	
Entradas	e1) Controle de Projetos e2) Relatório de Viabilidade do Projeto e3) Fluxo de caixa
Saídas	s1) Controle de Projetos s2) Cobrança s3) Fluxo de Caixa
Tarefas	<ul style="list-style-type: none"> - Dar baixa nas duplicatas pagas, na aba "Duplicatas" do arquivo "Controle de Projetos" - Confrontar relatório de duplicatas pagas com emitidas e cobrar os clientes que estão com suas duplicatas vencidas - Relatar à equipe de finanças a respeito do cliente inadimplente - Analisar "Relatório de Viabilidade do Projeto" <ul style="list-style-type: none"> * Inserir no "Fluxo de caixa" mensal os recebimentos * Confrontar o "Relatório de viabilidade do Projeto" com o "Fluxo de caixa" mensal * Tomar as decisões cabíveis * Atualizar "Fluxo de caixa"

QUADRO A3.23.8 – DETALHAMENTO DA ATIVIDADE E08

Atividade	E08. Conduzir revisão do milestone: Arquitetura do Software
Responsáveis	Equipe de Qualidade Gerente de Projeto
Descrição	Essa atividade tem o objetivo de revisar o milestone que marca o final dessa fase.
Recursos	
Entradas	e1) Plano de Garantia da Qualidade e2) Produto de Trabalho: Arquitetura do Software e3) Relatório sobre Garantia de Qualidade
Saídas	s1) Relatório sobre Garantia de Qualidade
Tarefas	<ul style="list-style-type: none"> - Realizar a atividade "GqPP01. Avaliar o processo e assegurar resolução de não conformidades" - Realizar a atividade "GqPP02. Avaliar os produtos de trabalho e assegurar resolução de não conformidades" - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

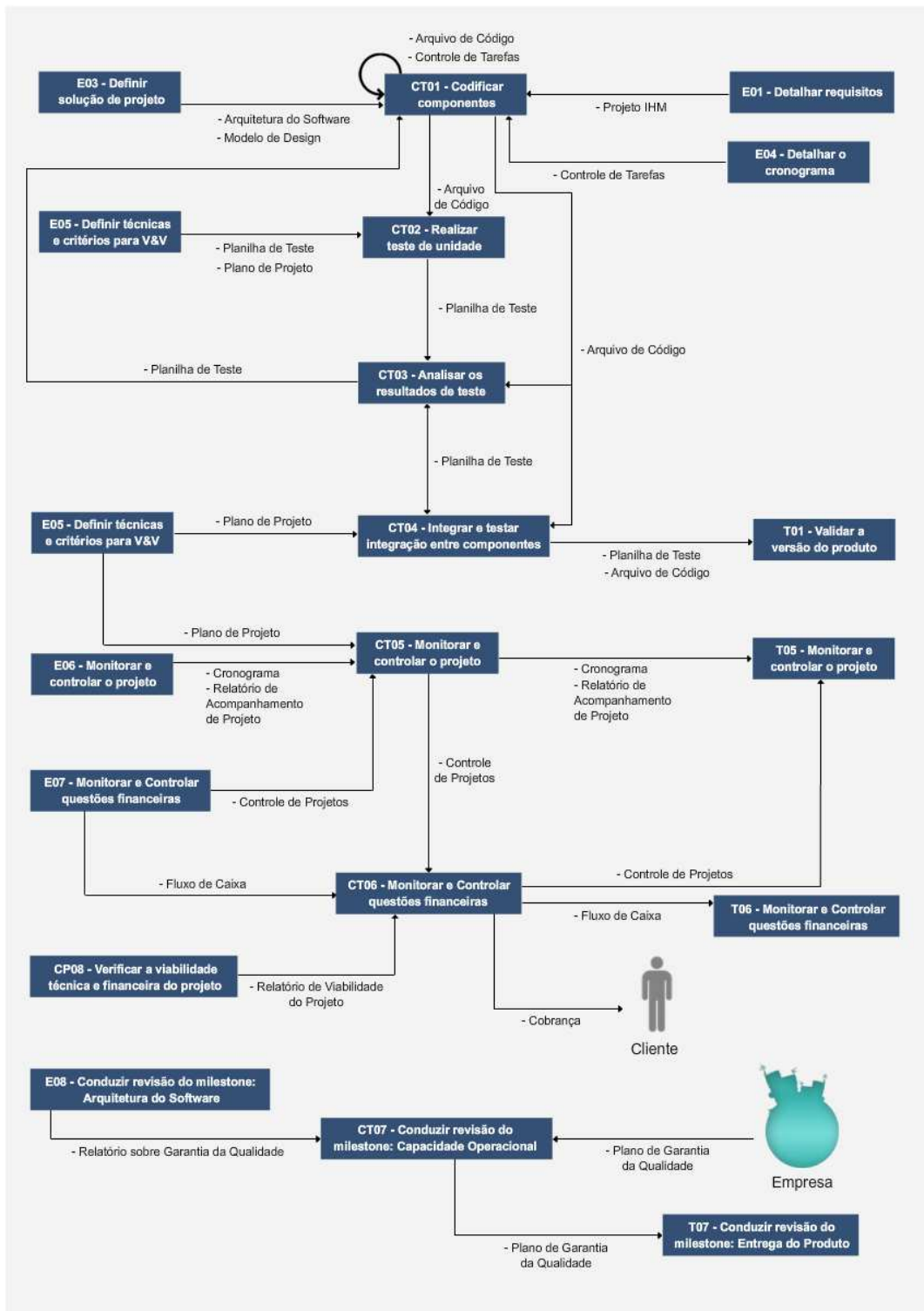


FIGURA A3.24 – VISÃO GRÁFICA – FASE: CONSTRUÇÃO

QUADRO A3.24.1 – DETALHAMENTO DA ATIVIDADE Ct01

Atividade	Ct01. Codificar componentes
Responsáveis	Equipe de Desenvolvimento
Descrição	Essa atividade tem por objetivo a codificação das partes que compõem o projeto em uma determinada linguagem de programação.
Recursos	Linguagem e Ambiente de Programação
Entradas	e1) Arquitetura do Software e2) Modelo de Design e3) Projeto IHM e4) Controle de Tarefas e5) Arquivo de Código e6) Planilha de Teste
Saídas	s1) Arquivo de Código s2) Controle de Tarefas
Tarefas	- Para cada versão planejada no ciclo de desenvolvimento: * Criar/Atualizar "Arquivo de Código" (conforme GCf01), com base nos artefatos: "Arquitetura do Software", "Modelo de Design", "Projeto IHM" e, se for o caso de ajustar o código, usar como subsídio os casos de teste marcados na "Planilha de Teste" * Realizar as tarefas de codificação, seguindo regras de padronização de codificação da empresa para que o código fique auto-documentado. * Registrar trabalho realizado e quantidade de tempo gasto no arquivo "Controle de Tarefas" para informações sobre o andamento de suas tarefas a serem comunicadas ao gerente de projeto, quando solicitado. - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

QUADRO A3.24.2 – DETALHAMENTO DA ATIVIDADE Ct02

Atividade	Ct02. Realizar teste de unidade
Responsáveis	Equipe de Desenvolvimento Equipe de Teste
Descrição	Essa atividade tem por objetivo executar os casos de teste de unidade definidos no plano de teste.
Recursos	- Ferramentas de Teste (ex: TestLink, Testitool, PHP Coverage, FunkLoad) - Linguagem e Ambiente de Programação
Entradas	e1) Plano de Projeto e2) Planilha de Teste e3) Arquivo de Código
Saídas	s1) Planilha de Teste s2) Arquivo de Código

QUADRO A3.24.2 – DETALHAMENTO DA ATIVIDADE CT02 (CONT.)

Tarefas	<p>- Realizar o Teste de Unidade planejado e registrado no "Plano de Projeto" das unidades codificadas</p> <ul style="list-style-type: none"> * Executar os casos de teste de unidade definidos na planilha "Casos de Teste de Unidade" do arquivo "Planilha de Teste" * Registrar resultados obtidos na na planilha "Casos de Teste de Unidade" do arquivo "Planilha de Teste" * Quando necessário, realizar novamente os casos de teste (teste de regressão), ou ainda, definir novos casos de teste e registrar na planilha "Casos de Teste de Unidade" do arquivo "Planilha de Teste" para testar as alterações realizadas no "Arquivo de Código" <p>- Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"</p>
----------------	---

QUADRO A3.24.3 – DETALHAMENTO DA ATIVIDADE CT03

Atividade	Ct03. Analisar os resultados de teste
Responsáveis	Equipe de Desenvolvimento Equipe de Teste
Descrição	Essa atividade tem por objetivo analisar os resultados de teste.
Recursos	
Entradas	e1) Arquivo de Código e2) Planilha de Teste
Saídas	s1) Planilha de Teste
Tarefas	<p>- Confrontar os resultados obtidos com os resultados esperados que se encontram na "Planilha de Teste" e marcar na planilha quais casos de teste detectaram problemas</p> <ul style="list-style-type: none"> - Fazer os ajustes necessários no "Arquivo de Código", acessando a atividade "Ct01. Codificar componentes" <p>- Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"</p>

QUADRO A3.24.4 – DETALHAMENTO DA ATIVIDADE Ct04

Atividade	Ct04. Integrar e testar integração entre componentes
Responsáveis	Equipe de Desenvolvimento Equipe de Teste
Descrição	Essa atividade tem por objetivo integrar os componentes e executar os casos de teste de integração definidos no plano de teste.
Recursos	- Ferramentas de Teste (ex: TestLink, Testitool, PHP Coverage, FunkLoad) - Linguagem e Ambiente de Programação
Entradas	e1) Arquivo de Código e2) Plano de Projeto e3) Planilha de Teste
Saídas	s1) Planilha de Teste s2) Arquivo de Código
Tarefas	- Realizar integração dos componentes, de acordo com a estratégia definida no "Plano de Projeto" * Averiguar se os componentes estão prontos para serem integrados (considerando não só os componentes de software, mas também de hardware se for o caso) - Realizar o Teste de Integração, conforme os componentes/módulos são integrados * Executar os casos de teste de integração definidos na planilha "Casos de Teste de Integração" do arquivo "Planilha de Teste", verificando a correteidade das interfaces que conectam os componentes * Registrar resultados obtidos na planilha "Casos de Teste de Integração" do arquivo "Planilha de Teste" - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

QUADRO A3.24.5 – DETALHAMENTO DA ATIVIDADE Ct05

Atividade	Ct05. Monitorar e controlar o projeto
Responsáveis	Gerente de Projeto
Descrição	Essa atividade tem como objetivo o monitoramento e controle do projeto.
Recursos	Ferramenta de gerenciamento de projeto (ex: dot Project, MS-Project, Primavera)
Entradas	e1) Plano de Projeto e2) Relatório de Acompanhamento de Projeto e3) Controle de Projetos e4) Cronograma
Saídas	s1) Relatório de Acompanhamento de Projeto s2) Controle de Projetos s3) Cronograma

QUADRO A3.24.5 – DETALHAMENTO DA ATIVIDADE CT05 (CONT.)

Tarefas	<ul style="list-style-type: none"> - Em qualquer iteração e ciclo de desenvolvimento: <ul style="list-style-type: none"> * Inserir a data de término da fase do projeto na planilha "Acompanhamento de Projetos" do arquivo "Controle de Projetos", replicar a linha do projeto e inserir a nova fase com sua data de início * Inserir na planilha "Acompanhamento de Projetos" do arquivo "Controle de Projetos" a quantidade de horas trabalhadas de cada um dos colaboradores nessa fase. * Inserir no "Cronograma" a quantidade de horas trabalhadas de cada um dos colaboradores nessa fase. * Verificar no "Cronograma" o andamento das atividades executadas em relação às atividades planejadas <ul style="list-style-type: none"> - Identificar e gerenciar dependências em relação às datas planejadas - Identificar compromissos que não foram satisfeitos - Identificar a ocorrência de algum risco levantado no "Plano de Projeto" e, nesse caso, executar o "Plano de Contingência" documentado no "Plano de Projeto" - Criar/Atualizar o arquivo "Relatório de Acompanhamento de Projeto" (conforme GCf01) e registrar a revisão de andamento do projeto nesse relatório * Planejar e executar ações corretivas, caso necessário, e registrá-las no "Relatório de Acompanhamento de Projeto" - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"
----------------	--

QUADRO A3.24.6 – DETALHAMENTO DA ATIVIDADE CT06

Atividade	Ct06. Monitorar e Controlar questões financeiras
Responsáveis	Gerente de Projeto Equipe de Finanças
Descrição	Essa atividade tem como objetivo o monitoramento e controle de todas as questões financeiras referentes ao projeto.
Recursos	
Entradas	e1) Controle de Projetos e2) Relatório de Viabilidade do Projeto e3) Fluxo de caixa
Saídas	s1) Controle de Projetos s2) Cobrança s3) Fluxo de Caixa
Tarefas	<ul style="list-style-type: none"> - Dar baixa nas duplicatas pagas, na aba "Duplicatas" do arquivo "Controle de Projetos" - Confrontar relatório de duplicatas pagas com emitidas e cobrar os clientes que estão com suas duplicatas vencidas - Relatar à equipe de finanças a respeito do cliente inadimplente - Analisar "Relatório de Viabilidade do Projeto" <ul style="list-style-type: none"> * Inserir no "Fluxo de caixa" mensal os recebimentos * Confrontar o "Relatório de viabilidade do Projeto" com o "Fluxo de caixa" mensal * Tomar as decisões cabíveis * Atualizar "Fluxo de caixa"

QUADRO A3.24.7 – DETALHAMENTO DA ATIVIDADE Ct07

Atividade	Ct07. Conduzir revisão do milestone: Capacidade Operacional
Responsáveis	Equipe de Qualidade
Descrição	Essa atividade tem o objetivo de revisar o milestone que marca o final dessa fase. É determinado se o produto está pronto para ser distribuído em um ambiente de teste beta.
Recursos	
Entradas	e1) Plano de Garantia da Qualidade e2) Produto de Trabalho: Arquivo de Código (Capacidade Operacional do Produto) e3) Relatório sobre Garantia da Qualidade
Saídas	s1) Relatório sobre Garantia da Qualidade
Tarefas	- Realizar a atividade "GqPP01. Avaliar o processo e assegurar resolução de não conformidades" - Realizar a atividade "GqPP02. Avaliar os produtos de trabalho e assegurar resolução de não conformidades" - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

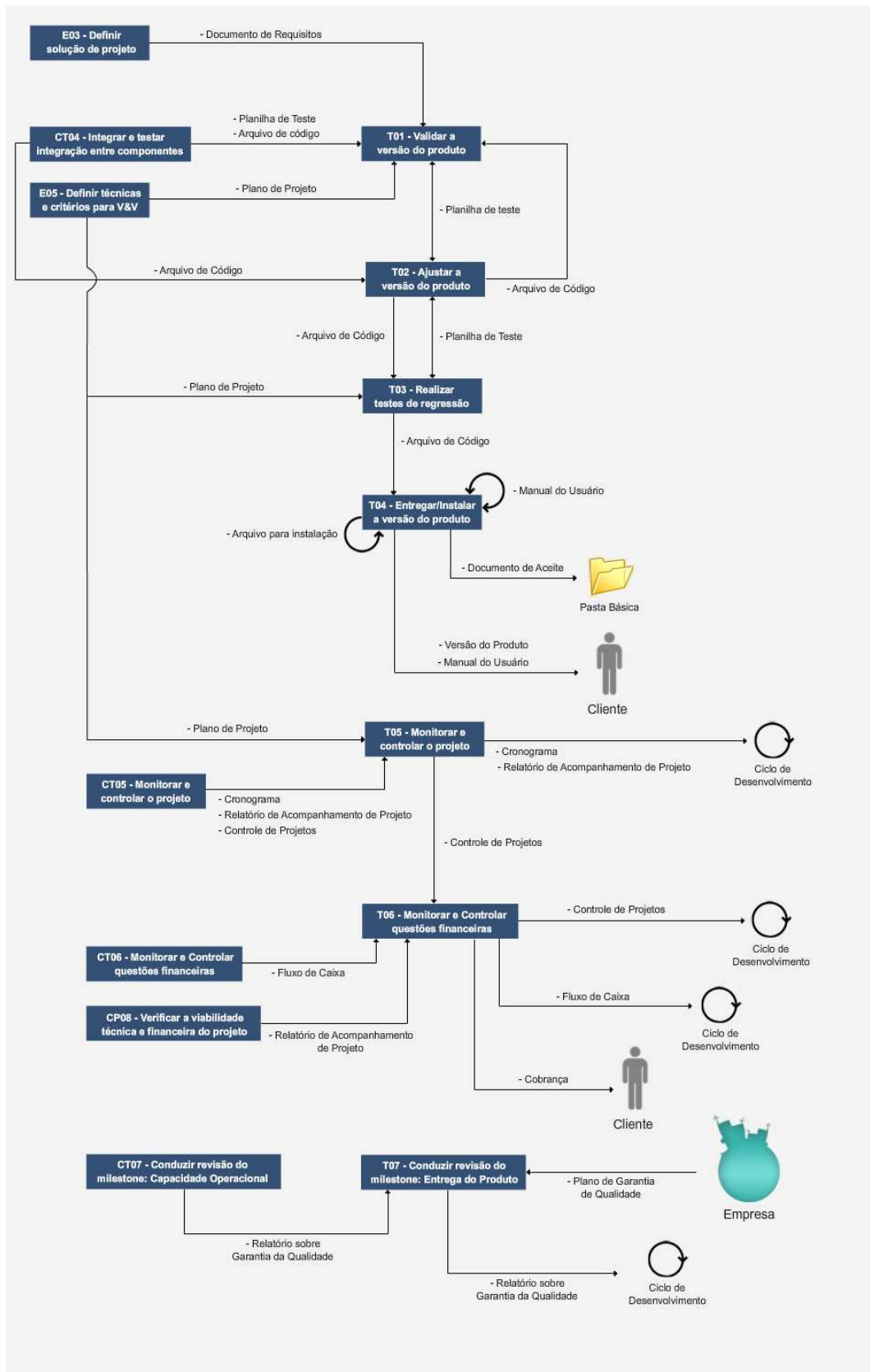


FIGURA A3.25 – VISÃO GRÁFICA – FASE: TRANSIÇÃO

QUADRO A3.25.1 – DETALHAMENTO DA ATIVIDADE T01

Atividade	T01. Validar a versão do produto
Responsáveis	Equipe de Desenvolvimento Equipe de Teste
Descrição	Essa atividade tem por objetivo executar os casos de teste de validação definidos no plano de teste na versão do produto.
Recursos	
Entradas	e1) Arquivo de Código e2) Plano de Projeto e3) Documento de Requisitos e4) Planilha de Teste
Saídas	s1) Planilha de Teste
Tarefas	- Realizar o Teste de Validação da versão do produto, simulando o ambiente do cliente * Executar os casos de teste de validação definidos na planilha "Casos de Teste de Validação" do arquivo "Planilha de Teste" * Registrar resultados obtidos na planilha "Casos de Teste de Validação" do arquivo "Planilha de Teste"

QUADRO A3.25.2 – DETALHAMENTO DA ATIVIDADE T02

Atividade	T02. Analisar e Ajustar a versão do produto
Responsáveis	Equipe de Desenvolvimento Equipe de Teste
Descrição	Essa atividade tem por objetivo analisar os resultados de teste realizar as modificações necessárias na versão do produto validado.
Recursos	Linguagem e Ambiente de Programação
Entradas	e1) Arquivo de Código e2) Planilha de Teste
Saídas	s1) Arquivo de Código s2) Planilha de Teste
Tarefas	- Confrontar os resultados obtidos com os resultados esperados que se encontram na "Planilha de Teste" - Confrontar os resultados obtidos em relação aos critérios de validação definidos "Documento de Requisitos" - Marcar na "Planilha de Teste" os critérios de validação que não foram atendidos - Fazer os ajustes necessários no "Arquivo de Código" - Validar o "Arquivo de Código" modificado, acessando a atividade "T01. Validar a versão do produto" - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

QUADRO A3.25.3 – DETALHAMENTO DA ATIVIDADE T03

Atividade	T03. Realizar testes de regressão
Responsáveis	Equipe de Desenvolvimento Equipe de Teste
Descrição	Essa atividade tem por objetivo executar novamente os casos de teste de unidade e de integração na versão ajustada do produto.
Recursos	- Ferramentas de Teste (ex: TestLink, Testitool, PHP Coverage, FunkLoad) - Linguagem e Ambiente de Programação
Entradas	e1) Plano de Projeto e2) Planilha de Teste e3) Arquivo de Código
Saídas	s1) Planilha de Teste
Tarefas	- Realizar, novamente, o Teste de Unidade, o Teste de Integração e o Teste de Validação planejados e registrados no "Plano de Projeto" das unidades codificadas anteriormente * Executar os casos de teste de unidade e os casos de teste de integração definidos nas planilhas "Casos de Teste de Unidade", "Casos de Teste de Integração" e "Casos de Teste de Validação" do arquivo "Planilha de Teste" * Registrar resultados obtidos nas respectivas planilhas - Realizar os ajustes necessários na versão do produto, se for o caso, acessando a atividade "T02. Ajustar a versão do produto"

QUADRO A3.25.4 – DETALHAMENTO DA ATIVIDADE T04

Atividade	T04. Entregar/Instalar a versão do produto
Responsáveis	Equipe de Desenvolvimento Equipe de Suporte Equipe de Vendas
Descrição	Essa atividade tem por objetivo a entrega e instalação (se for o caso) da versão do produto para uso no ambiente do cliente.
Recursos	
Entradas	e1) Arquivo de Código e2) Manual do Usuário
Saídas	s1) Versão do Produto s2) Documento de Aceite s3) Manual do Usuário s4) Arquivo para Instalação
Tarefas	<p>- Para documentar um release, deve-se:</p> <ul style="list-style-type: none"> * registrar as versões específicas dos componentes de código-fonte usados para criar o código executável * manter cópias dos códigos-fonte, do código-executável, de todos os arquivos de dados e de configuração * registrar as versões do sistema operacional, as bibliotecas, os compiladores e outras ferramentas usadas para construir o software <p>- Gerar a versão executável do produto, a partir do "Arquivo de Código"</p> <p>- Programar a implantação</p> <p>- Criar/atualizar o "Arquivo para Instalação" (conforme GCf01)</p> <p>- Realizar programação de implantação</p> <p>- Treinar o cliente</p> <p>- Criar/Atualizar o "Manual do Usuário" (conforme GCf01)</p> <p>- Criar/Atualizar arquivo "Documento de Aceite" (conforme GCf01) e coletar assinatura para confirmar que a versão foi entregue ao cliente. Anexar à pasta física.</p> <p>* Quando se tratar da versão final do sistema, o Manual do Usuário será entregue.</p> <p>- Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"</p>

QUADRO A3.25.5 – DETALHAMENTO DA ATIVIDADE T05

Atividade	T05. Monitorar e controlar o projeto
Responsáveis	Gerente de Projeto
Descrição	Essa atividade tem como objetivo o monitoramento e controle do projeto.
Recursos	Ferramenta de gerenciamento de projeto (ex: dot Project, MS-Project, Primavera)
Entradas	e1) Plano de Projeto e2) Relatório de Acompanhamento de Projeto e3) Controle de Projetos e4) Cronograma
Saídas	s1) Relatório de Acompanhamento de Projeto s2) Controle de Projetos s3) Cronograma
Tarefas	<p>- Em qualquer iteração e ciclo de desenvolvimento:</p> <ul style="list-style-type: none"> * Inserir a data de término da fase do projeto na planilha "Acompanhamento de Projetos" do arquivo "Controle de Projetos", replicar a linha do projeto e inserir a nova fase com sua data de início * Inserir na planilha "Acompanhamento de Projetos" do arquivo "Controle de Projetos" a quantidade de horas trabalhadas de cada um dos colaboradores nessa fase. * Inserir no "Cronograma" a quantidade de horas trabalhadas de cada um dos colaboradores nessa fase. * Verificar no "Cronograma" o andamento das atividades executadas em relação às atividades planejadas <ul style="list-style-type: none"> - Identificar e gerenciar dependências em relação às datas planejadas - Identificar compromissos que não foram satisfeitos - Identificar a ocorrência de algum risco levantado no "Plano de Projeto" e, nesse caso, executar o "Plano de Contingência" documentado no "Plano de Projeto" - Criar/Atualizar o arquivo "Relatório de Acompanhamento de Projeto" (conforme GCf01) e registrar a revisão de andamento do projeto nesse relatório * Planejar e executar ações corretivas, caso necessário, e registrá-las no "Relatório de Acompanhamento de Projeto" <p>- Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"</p>

QUADRO A3.25.6 – DETALHAMENTO DA ATIVIDADE T06

Atividade	T06. Monitorar e Controlar questões financeiras
Responsáveis	Gerente de Projeto Equipe de Finanças
Descrição	Essa atividade tem como objetivo o monitoramento e controle de todas as questões financeiras referentes ao projeto.
Recursos	
Entradas	e1) Controle de Projetos e2) Relatório de Viabilidade do Projeto e3) Fluxo de caixa
Saídas	s1) Controle de Projetos s2) Cobrança s3) Fluxo de Caixa
Tarefas	<ul style="list-style-type: none"> - Dar baixa nas duplicatas pagas, na aba "Duplicatas" do arquivo "Controle de Projetos" - Confrontar relatório de duplicatas pagas com emitidas e cobrar os clientes que estão com suas duplicatas vencidas - Relatar à equipe de finanças a respeito do cliente inadimplente - Analisar "Relatório de Viabilidade do Projeto" <ul style="list-style-type: none"> * Inserir no "Fluxo de caixa" mensal os recebimentos * Confrontar o "Relatório de viabilidade do Projeto" com o "Fluxo de caixa" mensal * Tomar as decisões cabíveis * Atualizar "Fluxo de caixa"

QUADRO A3.25.7 – DETALHAMENTO DA ATIVIDADE T07

Atividade	T07. Conduzir revisão do milestone: Entrega do Produto
Responsáveis	Equipe de Qualidade Gerente de Projeto
Descrição	Essa atividade tem o objetivo de revisar o milestone que marca o final dessa fase. É decidido se os objetivos do projeto foram alcançados, e se deve ser iniciado um outro ciclo de desenvolvimento (passagem pelas 4 fases; cada passagem produz uma versão do software; tipicamente, os ciclos tem as fases de concepção e elaboração mais curtas, uma vez que a definição e arquitetura do produto são determinados nos ciclos anteriores; exceções ocorrem qdo é necessária a redefinição do produto ou arquitetura).
Recursos	
Entradas	e1) Plano de Garantia da Qualidade e2) Produto de Trabalho: Entrega do Produto e3) Relatório sobre Garantia da Qualidade
Saídas	s1) Relatório sobre Garantia da Qualidade
Tarefas	- Realizar a atividade "GqPP01. Avaliar o processo e assegurar resolução de não conformidades" - Realizar a atividade "GqPP02. Avaliar os produtos de trabalho e assegurar resolução de não conformidades" - Obs: Para fazer alterações nos artefatos, acione a área de Gestão de Mudanças e Configuração, a partir da atividade "GCf03. Solicitar mudanças"

Apêndice 4 – Modelo ProcSoftVD (Templates)

Os documentos A4.1 a A4.34, apresentados a seguir, são referentes aos *templates* dos artefatos de entrada e saída utilizados/criados nas/pelas atividades do modelo ProcSoftVD.

A4.1 - CHECKLIST

(Material elaborado pela Profa. Sandra Fabbri – UFSCAR (Universidade Federal de São Carlos).

Obtido por meio da URL: <http://www2.dc.ufscar.br/~junia/Checklist.pdf> em 30/09/2008)

Checklist é uma técnica usada durante revisões técnicas formais (FTR), atividade que garante a qualidade do software. Seus objetivos são:

- Descobrir erros de função, de lógica ou de implementação para qualquer representação do software.
- Verificar que o software em questão atende aos requisitos especificados.
- Garantir que o software foi representado de acordo com padrões pré-definidos.
- Garantir que o software seja desenvolvido de maneira uniforme.
- Desenvolver projetos mais gerenciáveis.

O checklist ajuda o gerente a coordenar a reunião de FTR e ajuda cada revisor a focar nas características importantes. Os checklists devem ser aplicados a documentos de análise, projeto, codificação, e teste, focando os aspectos e defeitos comuns da fase correspondente. Por exemplo, as questões no checklist para a fase de requisitos devem apontar os problemas e defeitos que podem aparecer nos documentos de especificação de requisitos e correlatos.

Entrada: um conjunto de requisitos do novo sistema

Saída: uma lista de defeitos que devem ser corrigidos no novo sistema

Para o documento de requisitos, devem ser aplicadas as seguintes categorias de perguntas:

Questões Gerais:

1. Os objetivos do sistema foram definidos?
2. Os requisitos estão claros e sem ambigüidades?
3. É fornecida uma visão geral do sistema?
4. É fornecida uma visão geral das formas de operação do sistema?
5. O software e hardware necessários são especificados?
6. Se existem suposições que afetam a implementação, elas são especificadas?
7. Para cada função, os requisitos foram especificados em termos de entrada, processamento e saída?
8. Todas as funções, os dispositivos e as restrições são mapeados nos requisitos e vice-versa?

1. Omissão (O):

1. Funcionalidades:

- 1.1. As funções descritas são suficientes para atender os objetivos do sistema?
- 1.2. As entradas declaradas para cada função são suficientes para executar essa função?
- 1.3. Os eventos indesejados foram considerados e as respostas necessárias a eles são especificadas?
- 1.4. O estado inicial e estados especiais foram considerados? (por exemplo, inicialização do sistema, término finalização anormal, etc)

2. Desempenho:

- 2.1. O sistema pode ser testado, demonstrado, analisado ou inspecionado para mostrar que satisfaz os requisitos?
- 2.2. Os tipos de dados, unidades, limites e resolução foram especificados?
- 2.3. A frequência e volume de entrada e saída foram especificados para cada função foram especificadas?

3. Interface:

- 3.1. As entradas e saídas para todas as interfaces são suficientes?
- 3.2. Foram especificados os requisitos de interface entre hardware, software pessoas e procedimentos?

4. Ambiente:

- 4.1. Foram especificadas de forma apropriada as funcionalidades de interação entre hardware, software com o sistema?

2. Fato incorreto (FI):

- 2.1. Todas as funções descritas são necessárias para atingir os objetivos do sistema?
- 2.2. Todas as entradas da função, para cada função, são necessárias para executar a função?
- 2.3. As entradas e saídas para todas as interfaces são necessárias?
- 2.4. As saídas produzidas por uma função, para cada função, são usadas por outra função ou transferidas para a interface externa?

3. Inconsistência (I):

- 3.1. Os requisitos são consistentes entre si?
- 3.2. Os requisitos funcionais são consistentes com o sistema operacional?

4. Ambigüidade (A):

- 4.1. Cada requisito é definido de tal forma que seja discreto, sem ambigüidade, e testável?
- 4.2. Todas as transições de modo são especificadas de forma determinística?

5. Informações estranhas (IE):

- 5.1. As funções especificadas são coerentes com o sistema e com os objetivos a serem alcançados?

6. Miscelânea (M)

Nro Sequencial	Local no Doc. Requisitos	Tipo do Defeito	Descrição
1	Requisito funcional 3	FI	A informação de valor médio de compras do último semestre não é utilizada pro nenhuma outra função.
2	Requisito funcional 14	A	Não fica claro qual a taxa de juros deve ser aplicada, quando acontece o atraso de pagamento do boleto bancário

de acordo com a taxonomia de erros

nro da seção ou do requisito no doc. de requisitos

uma explicação que dê para entender porque o inspetor considera que aquilo seja um defeito

A4.2 – ARQUITETURA DO SOFTWARE

ID documento: Data: / / Versão :
Responsável pelo documento:

ID Projeto:

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

1 Fatores Arquiteturais

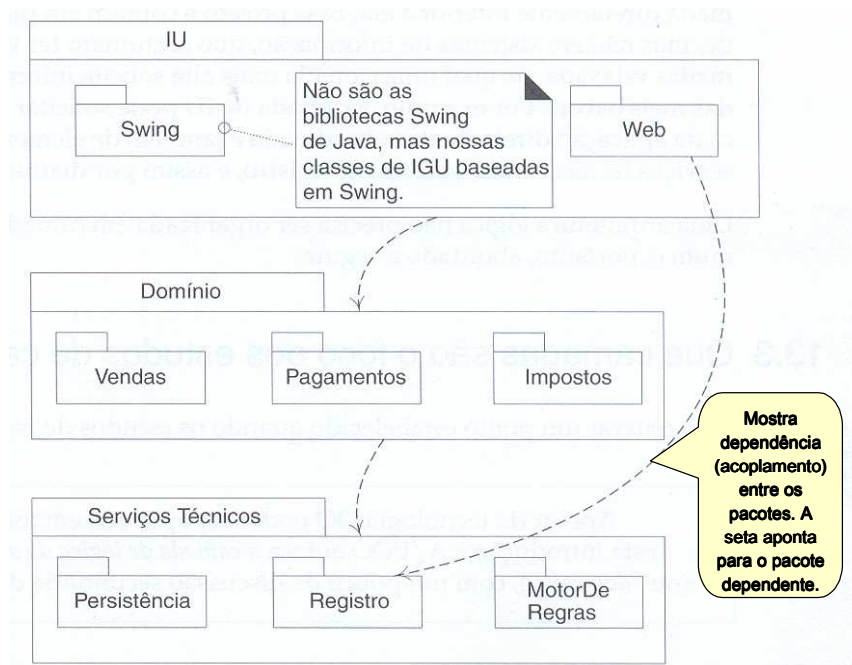
<Aqui são colocadas as referências relacionadas principalmente aos requisitos não funcionais que influenciarão na escolha da arquitetura.>

2 Arquitetura Lógica

< A arquitetura lógica é a organização em larga escala das classes de software em pacotes, subsistemas e, dependendo do estilo arquitetural, em camadas. É denominada arquitetura lógica porque não há decisão sobre como esses elementos são implantados pelos diferentes processos do sistema operacional ou pelos computadores físicos em uma rede (decisões essas posteriores, parte da arquitetura de implantação).

Uma camada é um agrupamento de granularidade muito grossa de classes, pacotes ou subsistemas que tem responsabilidade coesiva sobre um tópico importante do sistema.

Exemplo:



Fonte: LARMAN, C. *Utilizando UML e Padrões: Uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo*. Porto Alegre: Bookman, 2007.

>

3 Topologia do Sistema

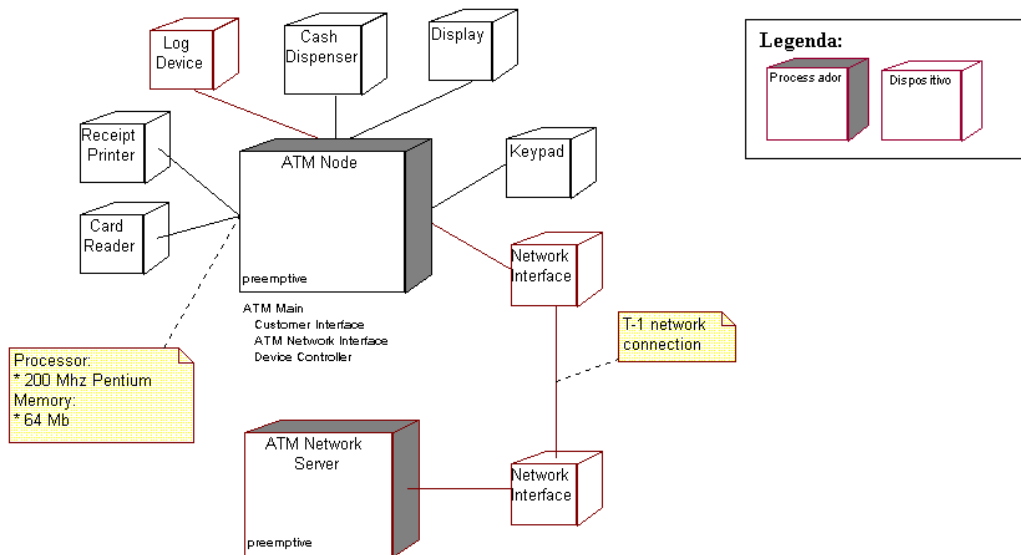
< A topologia do sistema mostra a implantação de elementos de software à arquitetura do software e a comunicação (geralmente em uma rede) entre elementos físicos.

A topologia do sistema pode ser representada por um diagrama de implantação da UML.

O Modelo de Implantação é opcional para sistemas de um único processador ou sistemas simples com pouca ou nenhuma distribuição de processamento. É obrigatório para sistemas com configurações complexas de rede ou de processador.

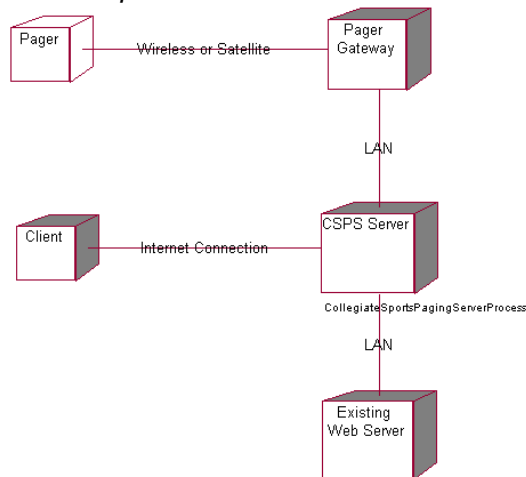
Um diagrama de implantação mostra a atribuição de artefatos concretos de software (como arquivos executáveis) a nós computacionais (algo com serviços de processamento).

Exemplo:



Fonte: RUP – Modelo pertencente à ferramenta CASE Rational Rose

Outro exemplo:



Fonte: Modelagem disponível (csps_design_v1_0.mdl) na ferramenta Rational Rose

>

4 Decisões Arquiteturais

<Descreve as decisões quanto à arquitetura e as razões técnicas que motivaram a tomada dessas decisões.>

<

Referência Bibliográfica: CLEMENTS, P.; BACHMANN, F.; BASS, L.; GARLAN, D.; IVERS, J.; LITTLE, R.; NORD, R.; STAFFORD, J. Documenting Software Architectures. Views and Beyond. SEI Series in Software Engineering. Addison Wesley, 2003.

A **arquitetura de software** é um conjunto de decisões significativas sobre a organização de um sistema, a seleção dos elementos estruturais e suas interfaces que compõem o sistema, junto ao seu comportamento especificado nas colaborações entre esses elementos, a composição desses elementos estruturais e

comportamentais em subsistemas mais abrangentes, e o estilo arquitetural que guia essa organização – esses elementos e suas interfaces, suas colaborações e sua composição (Booch et al., 1999).

Um tipo de visão define os tipos de elementos e os tipos de relacionamentos usados para descrever a arquitetura do sistema, a partir de determinada perspectiva.

Ao projetar um sistema, o arquiteto deve considerar três perspectivas do sistema:

1. Como o sistema é estruturado como um conjunto de unidades de implementação (tipo de visão módulo);
2. Como o sistema é estruturado como um conjunto de elementos que tem comportamento e interações em tempo de execução (tipo de visão componente-e-conector);
3. Como o sistema é relacionado às outras estruturas, exceto software, no seu ambiente (tipo de visão alocação).

Cada tipo de visão tem formas de ocorrer, denominadas estilos arquiteturais.

Um estilo arquitetural é uma especialização de tipos de elementos e relação, com um conjunto de restrições sobre como eles podem ser usados. Dessa forma, esses estilos refletem padrões de interação, independente do sistema em questão.

Nem todo sistema é construído exclusivamente a partir de um único estilo.

Normalmente, um sistema pode atender a vários estilos combinados.

Um módulo é uma unidade de código que implementa um conjunto de responsabilidades. Um módulo pode ser uma classe, uma coleção de classes, uma camada ou alguma decomposição da unidade de código. Todo módulo tem uma coleção de propriedades atribuídas a ele. Essas propriedades expressam informações importantes e restrições associadas ao módulo. Exemplos de propriedades: responsabilidades, visibilidade de informação e autor. Exemplos de relações entre os módulos: é parte de (is part of) ou herda de (inherits from).

Estilos arquiteturais do tipo de visão módulo:

- **Decomposição:** representa a decomposição do código em sistemas, subsistemas e assim por diante. Este estilo representa uma visão top-down do sistema.
- **Uso:** indica relacionamentos de dependência funcional entre os módulos. Esse estilo depende da relação uses, que é uma forma especial da relação depends-on. Esse estilo mostra aos desenvolvedores quais outros módulos devem existir para uma determinada parte do sistema executar de forma correta.
- **Generalização:** indica relacionamentos de especialização entre os módulos, como em uma hierarquia de classes. Esse estilo é normalmente utilizado para expressar design orientado a objetos.
- **Em camadas:** organiza o código em camadas disjuntas, de acordo com regras predefinidas. Por exemplo, regras podem estipular que somente o código da próxima camada mais abaixo pode ser usado. Indica a relação permissão-para-uso em uma parte restrita entre os módulos.

Estilos Arquiteturais do tipo de visão componente-e-conector (C&C) expressam comportamento em tempo real. Eles são descritos em termos de componentes e conectores. Um componente é uma das principais unidades de processamento da execução do sistema; um conector é um mecanismo de interação para os componentes. Objetos, processos ou coleções de objetos podem ser componentes. Conectores incluem pipes, repositórios e sockets. Middleware pode ser visualizado como um conector entre os componentes que usam o middleware. Componentes e conectores podem ser decompostos em outros componentes e conectores. Mais genericamente, visões arquiteturais em tempo de execução de

sistemas de objetos, tais como mostrado em diagramas de colaboração, são exemplos desse estilo de C&C. Alguns estilos de componentes-e-conectores são:

- *Pipe-and-filter*: é aquele no qual o padrão de interação é caracterizado pelas transformações sucessivas de dados. Dados chegam em um filtro, são transformados e passados pelo pipe (canal) para o próximo filtro no pipeline (canalização). Exemplos de sistemas desse tipo são sistemas de processamento de sinal e pipes UNIX.
- *Shared-data*: centraliza na retenção de dados persistentes. Vários elementos acessam os dados persistentes, os quais são retidos em pelo menos um repositório. Sistemas de base de dados e baseados em conhecimento são exemplos de estilos de dados compartilhados.
- *Publish-subscribe*: é caracterizado pelos componentes que interagem por eventos anunciados. Componentes podem subscrever para um conjunto de eventos. Esse estilo é comumente usado para dissociar produtores e consumidores de mensagens.
- *Client-server*: mostra componentes interagindo por meio de serviços solicitados de outros componentes. A essência desse estilo é que a comunicação é tipicamente emparelhada. Uma solicitação de serviço de um cliente é emparelhada com a provisão daquele serviço. Servidores fornecem um conjunto de serviços por meio de uma ou mais interfaces, e clientes usam zero ou mais serviços fornecidos pelos servidores no sistema. Pode haver um servidor central ou vários distribuídos.
- *Peer-to-peer*: é caracterizado pela interação de pares de componentes trocando serviços. É uma espécie de interação solicita/responde sem a assimetria encontrada no estilo cliente-servidor. Exemplos de sistemas peer-to-peer incluem arquiteturas que são baseadas em infra-estrutura de objetos distribuídos, tais como CORBA, COM+, Java RMI (remote method invocation).
- *Communicating-processes*: é caracterizado pela interação de componentes executados concorrentemente através de vários mecanismos de conexão. Exemplos desses mecanismos de conexão são: sincronização, passagem de mensagens, troca de dados, etc. Os processos de comunicação são comuns na maioria dos grandes sistemas e necessário em todos os sistemas distribuídos.

Os estilos do tipo de visão alocação descrevem o mapeamento das unidades de software com os elementos do ambiente (o hardware, sistema de arquivos ou equipe de desenvolvimento). São eles:

- *Implantação*: mapeia os processos para os elementos de hardware: nós de processamento, canais de comunicação, repositórios de memória e de dados. Esse estilo, usado para descrever como os processos são alocados para o hardware e o tráfego de mensagens resultante, é usado para análise de desempenho, segurança e confiabilidade e fornece uma base para estimar o custo de implantação de um único nó.
- *Implementação*: mapeia módulos de um tipo de visão módulos para uma infra-estrutura de desenvolvimento. Elementos do estilo de implementação são módulos e entidades de configuração. Esse estilo é usado para descrever como módulos são mapeados para entidades dentro do sistema de gestão de configuração, tanto quanto gerenciar versões e coordenar o desenvolvimento multi-equipes.
- *Designação de trabalho*: mapeia módulos de um tipo de visão módulos para equipes de desenvolvimento. Elementos desse estilo são módulos e equipes de desenvolvimento. O estilo é usado para descrever quais

equipes são responsáveis por quais elementos da WBS, tanto quanto para informar cronograma e estimativas de orçamento.

>

<

Referência Bibliográfica: SOMERVILLE, I. Engenharia de Software. 8ª edição. São Paulo: Addison Wesley, 2003.

A arquitetura afeta o desempenho, proteção, disponibilidade e facilidade de manutenção de um sistema. O estilo e a estrutura escolhidos para uma aplicação podem, portanto, depender dos requisitos não funcionais do sistema. Por exemplo:

- Quanto ao desempenho: se o desempenho for um requisito crítico, a arquitetura deve ser projetada para localizar operações críticas dentro de alguns subsistemas, com tão pouca comunicação quanto possível entre eles. Isso pode significar o uso de componentes de alta granularidade em detrimento dos de baixa granularidade para reduzir as comunicações entre os componentes.
- Quanto à proteção: se a proteção for um requisito crítico, uma estrutura em camadas para a arquitetura deve ser usada, com os itens mais críticos protegidos por camadas mais internas e com alto nível de validação de proteção aplicado a essas camadas.
- Quanto à segurança: se a segurança for um requisito crítico, a arquitetura deve ser projetada de modo que as operações relacionadas à segurança estejam todas localizadas em um único subsistema ou em um pequeno número de subsistemas. Isso reduz os custos e os problemas de validação de segurança e torna possível fornecer esse serviço a sistemas de proteção relacionados.
- Quanto à disponibilidade: se a disponibilidade for um requisito crítico, a arquitetura deve ser projetada para incluir componentes redundantes e, assim que possível, substituir e atualizar componentes sem parar o sistema.
- Quanto à facilidade de manutenção: se a facilidade de manutenção for um requisito crítico, a arquitetura de sistema deve ser projetada usando componentes de baixa granularidade e autocontidos que possam ser prontamente mudados. Os criadores de dados devem ser separados dos clientes, e estruturas de dados compartilhadas devem ser evitadas.

A arquitetura de um sistema pode ser baseada em um modelo ou estilo de arquitetura específico. Entretanto, a arquitetura da maioria dos grandes sistemas não está de acordo com um único estilo. Partes diferentes do sistema podem ser projetadas usando estilos diferentes de arquitetura. Por exemplo, um sistema pode ser organizado com base em um repositório de dados compartilhados, mas pode criar camadas com base nisso para apresentar uma visão mais abstrata dos dados.

>

A4.3 – CONTRATO

< Este contrato pode ser utilizado em casos de licença de software, incluindo o desenvolvimento, implantação e treinamento do sistema em questão e a entrega da documentação e programas fonte do sistema ao CONTRATANTE. Este template de contrato é uma adaptação de dois outros contratos: um cedido por Rogério Marcus Alessi (Secretário Municipal de Tecnologia da Informação da Prefeitura Municipal de Presidente Prudente, de 2004 a 2008) e outro disponibilizado na URL:

< http://www.ramosdainformatica.com.br/files/Clientes/modelo_contrato_desenvolvimento.pdf>

CONTRATO DE LICENÇA DE SOFTWARE

Pelo presente instrumento, de um lado <empresa>, inscrita no CNPJ sob número xxxxxxxx, sito à <endereço, nr>, cidade de xxxxxxxx, Estado de xxxxx, neste ato representada pelo(a) Sr(a). xxxx, CPF xxxxxxxxxx, doravante denominada **CONTRATANTE**, e de outro Sr(a). xxxxxxxx, sito à <endereço, nr>, cidade de xxxxxxxx, Estado de xxxxx, CPF xxxxxxxx, denominado **CONTRATADO**, têm entre si justo e contratado o que segue:

1. Objeto

1.1 O presente contrato tem como objeto o desenvolvimento, implantação e treinamento do <nome do sistema>, doravante denominado simplesmente “Sistema”, por parte da **CONTRATADA**, para uso específico da **CONTRATANTE**, conforme proposta técnica no Anexo 1, parte integrante deste instrumento.

2. Aspectos Técnicos

2.1 O prazo para desenvolvimento do sistema obedecerá ao cronograma no Apêndice 1, parte integrante deste instrumento.

2.2 O desenvolvimento e acompanhamento do sistema dar-se-á conforme estabelecido no cronograma citado no item 2.1, abrangendo reuniões e avaliações dos usuários da **CONTRATANTE** para desenvolvimento do Sistema.

2.3 <deixar claro sobre o suporte ao sistema, se está incluso no contrato ou não>

2.4 <deixar claro sobre as condições para instalação e uso do sistema>

3. Aspectos Financeiros

3.1 Pelo objeto proposto, obriga a **CONTRATANTE** a pagar à **CONTRATADA** a importância de R\$ xxxxxx (valor por extenso), de acordo com o cronograma de pagamentos no Apêndice 2, parte integrante deste instrumento.

3.2 Os custos adicionais com visitas, transporte e estadia, desde que previamente autorizados, serão reembolsados da seguinte forma: xxxxxxxxxxxx.

- 3.3 Não será cobrado da **CONTRATANTE** qualquer custo adicional referente a correções e ajustes efetuados no Sistema, salvo se decorrentes de implementações adicionais específicas e devidamente aprovadas por escrito pela **CONTRATANTE**, como: acréscimos de relatórios, módulos ou configurações solicitadas pela **CONTRATANTE** à **CONTRATADA**.
- 3.4 Qualquer uma das partes poderá rescindir o presente instrumento, mediante prévia comunicação à outra parte, por escrito, com antecedência mínima de **xx** dias, nas seguintes condições: **<especificar multa e condições de pagamento dos serviços até então executados e outras condições, como por exemplo, se a CONTRATADA poderá ficar ou não com o Sistema>**.

4. Aspectos Legais

- 4.1 O Sistema possui garantia vitalícia de funcionamento nas versões e condições (ambiente hardware/software) originalmente implantadas.
- 4.2 A responsabilidade da **CONTRATADA** restringir-se-á ao Sistema, não respondendo por problemas relacionados ao ambiente, como redes, sistemas operacionais e hardware.
- 4.3 A **CONTRATADA** não se responsabiliza por danos decorrentes do mau uso do sistema, alimentação errônea e/ou falta de conferência de dados gerados, bem como a inexistência de cópias de segurança dos dados atualizados.
- 4.4 O presente instrumento não dará azo à constituição de qualquer vínculo empregatício ou responsabilidade por parte da **CONTRATANTE** com relação aos empregados da **CONTRATADA** a seu serviço, responsabilizando-se esta última pelos direitos e deveres sociais e trabalhistas de seus empregados.
- 4.5 Pelo presente instrumento o **CONTRATANTE** receberá os programas fontes após sua regular quitação, responsabilizando-se pelo uso e guarda, nos termos da Lei 9.609/98.

5. Controle de Alterações

- 5.1 **<descrever os procedimentos de mudanças e a limitação para alteração dos requisitos.>**

E por estarem assim justas e contratadas, as partes assinam o presente contrato em **xx (nr de vias por extenso)** vias de igual teor e forma na presença das testemunhas, a seguir.

Cidade, dia de mês por extenso de ano.

CONTRATANTE

CONTRATADA

Testemunhas:

1ª) Ass. _____

Nome:

RG:

2ª) Ass. _____

Nome:

RG:

ANEXO 1 - PROPOSTA TÉCNICA: XXXXXXXXX

APÊNDICE 1 – CRONOGRAMA

<inserir aqui o cronograma, a partir da WBS>

APÊNDICE 2 – CRONOGRAMA DE PAGAMENTOS

A4.4 - DOCUMENTO DE REQUISITOS

ID documento: Data: / / Versão :

Responsável pelo documento:

ID Projeto:

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

1 – INTRODUÇÃO

1.1 Objetivo

Este documento tem por objetivo apresentar os requisitos que o sistema deve atender em diferentes níveis de detalhamento. Dessa forma, serve como acordo entre as partes envolvidas – cliente e analista/desenvolvedor.

1.2 Escopo

<

- Identificar o(s) produto(s) de software a ser(em) produzido(s) pelo nome.
- Explicar **o quê** o(s) produto(s) de software fará(ão) e, se necessário, o **quê** não fará(ão).
- Descrever a aplicação do software a ser especificado, incluindo benefícios relevantes, objetivos e metas.
- Ser consistente com as especificações de mais alto nível (tal como a especificação de requisitos do sistema), se existirem.>

1.3 Definições, Siglas e Abreviações

<Fornecer as definições de todos os termos, acrônimos e abreviações necessárias para interpretar de modo apropriado a ERS. >

1.4 Referências

<

- Fornecer uma lista completa de todos os documentos referenciados na ERS.
- Identificar cada documento pelo título, número do relatório (se aplicável), data e organização que publicou.
- Especificar a(s) origem(s) das referências, ou seja, onde e/ou com quem podem ser obtidas.>

1.5 Visão Geral

O Capítulo 2 fornece uma descrição geral do produto, tendo como público-alvo os clientes. Dessa forma, esse capítulo é uma síntese dos requisitos que o sistema deverá atender para auxiliar ao negócio do cliente. São descritos: a perspectiva e funções do produto, as características dos usuários e os limites, suposições e dependências que influenciem a eficácia e eficiência do sistema.

No Capítulo 3, os requisitos descritos no capítulo 2 são detalhados ao ponto de serem úteis para os analistas e programadores do sistema.

2 – DESCRIÇÃO GERAL DO PRODUTO

<Tem por objetivo descrever fatores gerais do produto e seus requisitos, fornecendo um contexto para esses requisitos os quais são definidos em detalhes no capítulo 3 da ERS.>

2.1 Perspectiva do Produto

*<Deve ser descrito de maneira **resumida**, de forma textual, sem detalhamento (1/2 página, no máximo, pois trata-se de uma descrição geral), pois as interfaces mencionadas nessa seção serão detalhadas na seção 3.1 (Requisitos de Interface Externa). Segundo o padrão IEEE, se o produto a ser desenvolvido for parte de um sistema maior a ERS deverá identificar quais as interfaces existentes entre esse sistema e o produto a ser desenvolvido.>*

Exemplo 1:

O sistema funcionará em um PC AT atualmente disponível na Locadora Fulano de Tal. O sistema terá interface com leitores de códigos de barras para simplificar o processo de alugar e devolver uma fita, e com impressoras do tipo tal para emitir os recibos para os clientes e para a própria locadora. Todas as informações relativas aos clientes, tais como: x, y, z; e informações históricas das locações serão armazenadas.

O texto pode incluir (não obrigatoriamente, pois depende do caso) informações sobre:

-Interfaces do Sistema: Normalmente um software faz parte de um sistema (sistema administrativo) maior existente dentro de uma empresa. A ERS deve listar as interfaces do sistema para com o produto, identificando as funcionalidades do software que irão realizar essas interfaces.

-Interfaces do Usuário: Características lógicas das interfaces entre o produto e seus usuários, como por exemplos: formatos de telas, aspectos de otimização da interface com o usuário do sistema (por ex, mensagens curtas ou longas, definir que um usuário pode utilizar o sistema após x horas de treinamento), padrão de relatórios ou menus de consulta, acesso por níveis de usuários, mensagens, dentre outros.

-Interfaces de Hardware: se o produto interage com dispositivos de hardware, estes devem ser especificados (por exemplo, impressoras, scanners, relógios de ponto, catracas eletrônicas ou outros dispositivos eletrônicos com o qual o produto irá comunicar-se).

-Interfaces de Software: especificar o uso de outros softwares necessários (banco de dados, sistemas operacionais, software para capturar imagem, ou outros softwares aplicativos de mesma natureza).

-Interfaces de Comunicação: especificar tipos de comunicação necessários para o funcionamento do produto. Por exemplo, o software que é responsável pelo gerenciamento da catraca precisa comunicar-se com as mesmas por meio de um par-trançado. Como, então, deverá ser implementada essa comunicação? Isso deve ser descrito aqui sem detalhes.

-Limites de Memória: especificar os limites mínimos de memória primária e secundária.

-Operações: rotinas de inicialização (definir níveis de acesso; processamento; backup e restauração do sistema).

-Requisitos para adaptação de situação: especificar situações em que o software deve ser adaptado antes da instalação. Por exemplo: em um sistema que necessite a conexão com a internet. Se no momento da instalação o computador não estiver conectado, o sistema identifica e grava os dados em um arquivo temporário e, quando restabelecer a conexão, os dados são recuperados deste arquivo temporário e a instalação pode concluída. Outro exemplo refere-se às adaptações necessárias para a instalação do software em outra versão do S.O.

>

2.2 Funções do Produto

< Nessa seção deve ser fornecido um resumo das principais funções que o software deve realizar. Além disso, pode ser inserido algum diagrama, tal como Diagrama de Casos de Uso, para mostrar a fronteira do sistema e para fornecer uma visão geral do comportamento do sistema (para que ele é usado e por quem).

Por exemplo:

O Sistema de Locadora de Vídeo deve manter os dados dos clientes, dos DVD's comprados de fornecedores registrados e das locações e devoluções realizadas por cada um dos clientes. Deve, também, permitir que o cliente faça a reserva de filmes, deve manter dados das contas a pagar e a receber e permitir a emissão do cupom fiscal.

Diagrama de Casos de Uso



>

2.3 Características do Usuário

<

Descrever o nível educacional dos usuários do sistema, bem como a sua experiência e o conhecimento sobre informática para que seja diagnosticada a necessidade de treinamento específico. Deve fornecer as razões pelas quais certos requisitos são especificados.>

2.4 Limites, Suposições e Dependências

<

Descrever itens que limitem as opções do desenvolvedor, tais como: Normas regulamentadoras, Limitações do hardware, considerações de segurança, requisitos de confiabilidade, Linguagem de programação.

Com relação às suposições e dependências, descrever qualquer fator que afeta os requisitos expressos na ERS. Por exemplo: A não aquisição do ponto eletrônico fará com que o sistema não tenha o seu total desempenho, pois a entrada de dados será feita manualmente, inserindo somente as exceções do ponto diário, ou seja, a falta dos funcionários.>

2.5 Requisitos Adiadados

<

Identificar os requisitos, que foram levantados, mas que por alguma razão serão adiados para versões futuras do sistema.>

3 – REQUISITOS ESPECÍFICOS

<Essa seção deve conter todos os requisitos do software com um nível de detalhamento suficiente para possibilitar aos projetistas/desenvolvedores projetarem um sistema que atenda a esses requisitos.>

3.1 Requisitos de Interface Externa

<Detalhar o que foi descrito de forma sucinta na seção 2.3 (perspectiva do produto) com relação às interfaces externas, sem repetir informação. Esses requisitos referem-se aos requisitos não funcionais >

3.1.1 Interfaces do Sistema

<caso o software em questão tenha que ser integrado com algum já existente>

3.1.2 Interfaces do Usuário

<Deve ser descrito como o usuário vai interagir com o sistema, sem mostrar graficamente as telas, pois existe uma seção específica para isso. Descrever como será o formato padrão das telas e relatórios, quais os procedimentos a serem adotados em caso de erros, para que servem e como serão apresentadas as mensagens do sistema para o usuário (por exemplo, não serão exibidas mensagens em caixas de diálogo, mas através de um label em um determinado local da tela. As mensagens exibidas em caixas de diálogos serão somente para erros do sistema que só podem ser resolvidas pelo desenvolvedor).>

3.1.3 Interfaces de Software

<descrever os detalhes dos softwares necessários para o desenvolvimento e execução do software em questão. Isso inclui nome, mnemônico, especificação numérica, versão e fonte>

3.1.4 Interfaces de Hardware

<Com relação à interface de **hardware**, por exemplo, a ERS deverá detalhar como será realizada a interface em questão. Se for um relógio de ponto, por exemplo, como será o layout do arquivo que será emitido pelo equipamento ao sistema. Por exemplo: O equipamento para a leitura do ponto dos funcionários será da marca XXXX, modelo YYYY, etc. O relógio ponto gera um arquivo com a extensão txt, o qual possui a seguinte estrutura em cada uma das linhas:

CÓDIGO_FUNCIONÁRIO: 4 caracteres, DATA: 10 caracteres, HORÁRIO: 5 caracteres

Ex.:

CODFUN	DATA	HORÁRIO
43219	05/09/2003	7:30
43219	05/09/2003	11:30
	...	

>

3.1.5 Interfaces de Comunicação

<Especificar os tipos de comunicação utilizados para integração com outros periféricos e tecnologias. Ex: par-trançado, protocolos de comunicação, etc>

3.2 Requisitos de Desempenho

<Especificar os requisitos numéricos estáticos e dinâmicos sobre o software ou uma interação humana com o software. Os requisitos numéricos estáticos podem incluir: o número de terminais suportado, o número de usuários simultâneos suportado, a quantidade e o tipo de informação a ser manipulado. Os requisitos numéricos dinâmicos podem incluir, por exemplo, o número de transações e tarefas e a quantidade de dados a ser processado dentro de determinado período de tempo e condições de pico de sobrecarga. Todos esses requisitos devem ser declarados em termos mensuráveis. Por ex: "95 % das transações devem ser processadas em menos de 1 segundo".>

3.3 Outros Requisitos

<Descrever aqui restrições do projeto que podem ser impostas por conformidade a padrões, limitações de hardware, e por outros requisitos não funcionais: manutenibilidade, portabilidade, confiabilidade, requisitos éticos, requisitos de entrega, etc.

>

3.4 Funções

<

Serão descritas todas as funções do produto. Esses requisitos funcionais podem ser representados por meio de texto estruturado em linguagem natural, mas também podem ser representados por meio de casos de uso, dentre outras maneiras.

A seguir, serão apresentadas duas alternativas para documentar os requisitos.

Alternativa 1) As funções são descritas por meio de um texto estruturado em linguagem natural e para cada função são descritos os itens de entrada necessários (dados/informação) e os itens de saída gerados, além das regras de negócio que irão influenciar as funções.

Essas funções podem ser classificadas em:

- **Funções Básicas:** referem-se às operações CRUD (create, read, update, delete) necessárias para a execução das funções fundamentais. Esse conjunto de operações pode ser denominado “Gerenciar dados de”.
- **Funções Fundamentais:** referem-se às transações de negócio (movimentações), que realmente agregam valor ao negócio;
- **Funções de Saída:** referem-se às funções que geram informações de saída relevantes para atender às necessidades do cliente (por exemplo, relatórios com cruzamento de informações). Nesse caso, devem ser descritos não só os itens de entrada (filtros), mas também os itens de saída (informações) pertinentes.

Observações:

- 1) é importante que cada função tenha um identificador, a fim de facilitar a rastreabilidade desse requisito. Sugere-se que seja utilizado RF (requisito funcional) seguido de um underline, uma letra indicando se é função básica, fundamental ou saída externa (B, F, S) e um número sequencial. Ex: RF_B1. e RF_B2. para funções básicas, RF_F1., RF_F2. para funções fundamentais e RF_S1., RF_S2. para funções de saída externa).
- 2) não devem ser citados aqui os campos das possíveis tabelas do sistema, tais como, códigos sequenciais criados para facilitar na implementação. Aqui deverão ser citados apenas os **itens de informação** relacionados às funções do sistema.
- 3) As funções de gerenciamento do usuário, backup e restauração do sistema não serão citadas aqui, uma vez que já foram descritas no item 2.3 – Perspectiva do Produto.

EXEMPLO: Em um sistema de locadora de vídeo:

FUNÇÕES BÁSICAS

RF_B1. Gerenciar cliente: o usuário pode inserir, consultar, alterar e deletar os dados pessoais do cliente (nome, endereço, cep, cidade, estado, CPF, data de nascimento, e-mail e fone para contato).

RF_B2. Gerenciar vídeo: o usuário pode inserir, consultar, alterar e deletar os dados relacionados aos vídeos (código do vídeo, título, gênero, quantidade, preço de locação).

FUNÇÕES FUNDAMENTAIS

RF_F1. Efetuar Reserva: o cliente pode fazer a reserva de determinado vídeo. Para isso são necessários os seguintes itens de informação: dados pessoais do cliente, dados do vídeo, data e hora da reserva. Caso o cliente ainda não esteja cadastrado no sistema, é necessário realizar um cadastro mesmo que somente com os itens obrigatórios: nome, CPF e fone.

RF_F2. Efetuar Locação: o cliente pode locar um vídeo, caso este não esteja reservado. São necessários os itens de informação: dados pessoais do cliente, dados do vídeo, preço da locação, data da locação e data para devolução (o cliente pode devolver o vídeo sem adicionais ao preço da locação em até 3 dias, após a data da locação). O registro da locação deve ser vinculado à uma conta a receber.

RF_F3. Efetuar Devolução: no ato da devolução são necessários os itens de informação: dados do cliente, dados do vídeo e data de devolução. Caso a data de devolução tenha ultrapassado os 3 dias após a locação, deve ser calculada uma multa de 10% sobre o valor da locação por dia de atraso.

RF_F4. Dar Baixa das contas a receber: o cliente pode optar por efetuar o pagamento no ato da locação ou da devolução. Sendo assim, deve ser registrada a data do pagamento e o valor pago, e deve ser gerado um cupom fiscal contendo as informações pertinentes à locação e ao pagamento.

RF_F5. Comprar vídeos por parte da locadora (incluindo contas a pagar):

RF_F6. Dar Baixa das contas a pagar:

FUNÇÕES DE SAÍDA

RF_S1. Listagem dos Clientes que mais locaram em determinado período: o usuário entra com o período e como saída tem-se uma lista contendo o nome, telefone de contato e e-mail de todos os clientes que mais locaram.

RF_S2. Balancete do mês:...

RF_S3. Fila de espera referente à reserva: ...

RF_S4. Listagem de Clientes inadimplentes: ...

Para complementar o entendimento dos requisitos, sugere-se elaborar o **Modelo Conceitual**, ou também denominado Modelo de Domínio, um modelo que pode ser utilizado como preliminar para a elaboração futura do modelo de dados do sistema. Esse tem por objetivo a visualização dos conceitos do domínio. Para a elaboração do *modelo conceitual utiliza-se da representação do diagrama de classes da UML, entretanto, são colocados somente o nome do conceito, os seus atributos mais relevantes e as multiplicidades. É importante ressaltar que um conceito, não necessariamente, será uma classe de implementação.*

Alternativa 2) *Elaborar uma Lista de Funcionalidades na qual serão descritas todas as funções do sistema (requisitos funcionais), detalhadamente, inclusive as operações CRUD (que não serão consideradas CDU).*

Na coluna **Referência** deve ser colocado um identificador corresponde ao requisito funcional. Exemplo: RF_1, RF_2. A coluna **Categoria** deve ser preenchida com **Evidente** (a função deve ser executada, e o usuário deve estar ciente da execução. Ex.: Registrar Venda, Processar Pagamento) ou **Oculto** (deve ser executada, mas de modo transparente para o usuário. Ex.: Dar baixa na qtde de um produto no estoque)

Exemplo:

Referência	Funções	Categoria
RF_B1	Gerenciar cliente	Evidente
RF_F1	Efetuar Reserva	Evidente
RF_F1.1	Verificar cadastro do cliente	Oculto
RF_F1.2	Registrar reserva	Oculto

A seguir, realizar as Especificações dos Casos de Uso Essenciais (sem definir considerações tecnológicas). Cada caso de uso (CDU) pode ser especificado usando um template (tal como o definido no RUP e mostrado, a seguir).

Seção do CDU	Comentário
Nome do Caso de Uso	Começar com um verbo.
Breve Descrição	Descrição em alto nível do CDU.
Fluxo Básico	Um caminho típico, incondicional e otimista do cenário de sucesso.
Fluxos Alternativos	Cenários Alternativos de sucesso ou fracasso.
Requisitos Especiais	Requisitos não funcionais relacionados.
Pré-Condições	O que precisa ser verdade antes da realização do CDU.
Pós-Condições	O que precisa ser verdade quando da finalização bem sucedida.
Relacionamentos	Os relacionamentos que envolvem os CDUs, tais como include e extend.

Ainda, para os CDUs que possuem atividades concorrentes, pode-se elaborar um Diagrama de Atividades.

Para complementar o entendimento dos requisitos, sugere-se elaborar o **Modelo Conceitual**, ou também denominado Modelo de Domínio, um modelo que pode ser utilizado como preliminar para a elaboração futura do modelo de dados do sistema. Esse tem por objetivo a visualização dos conceitos do domínio. Para a elaboração do *modelo conceitual utiliza-se da representação do diagrama de classes da UML, entretanto, são colocados somente o nome do conceito, os seus atributos mais relevantes e as multiplicidades. É importante ressaltar que um conceito, não necessariamente, será uma classe de implementação.*

>

VOLATILIDADE DOS REQUISITOS

RQ	Alta	Média	Baixa
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

MATRIZ DE RASTREABILIDADE - REQUISITOS CLIENTE X PRODUTO

RQC RQ							
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

MATRIZ DE RASTREABILIDADE - REQUISITOS FUNCIONAIS E NÃO FUNCIONAIS DO PRODUTO

RQC RQ							
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

MATRIZ DE RASTREABILIDADE - REQUISITOS DO PRODUTO X COMPONENTES

Com RQ							
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

APÊNDICE 1 - LISTA DE REQUISITOS DO CLIENTE

<Aqui são descritos os requisitos do cliente, da forma como eles os expõe. Muitas vezes, um requisito de cliente transforma-se em N requisitos do sistema.

1. <requisito 1 do cliente>
2. <requisito 2 do cliente >
3. <requisito 3 do cliente>
4. ...

>

APÊNDICE 2 - COMPONENTES EMPREGADOS

<Aqui são identificados os componentes utilizados no projeto, inclusive os já existentes em uma Biblioteca.

1. <componente 1>
2. <componente 2>
3. <componente 3>
4. ...

>

APÊNDICE 3 - PROTÓTIPOS

<Aqui são inseridos os protótipos, caso tenham sido construídos para auxiliar no levantamento e análise dos requisitos>

A4.5 - DOCUMENTO DE ACEITE

ID documento: **Data:** / / **Versão :**

Responsável pelo documento:

ID Projeto:

Nome	Cargo / Papel	Instituição	Aprovação	Observações	Data	Versão Avaliada	ID Comprovante assinado
			Aprovado <input type="checkbox"/> Reprovado <input type="checkbox"/> Aprovado com ressalvas <input type="checkbox"/> Nesse caso, data limite: dd/ mm/ aaaa		dd/ mm/ aaaa		
			Aprovado <input type="checkbox"/> Reprovado <input type="checkbox"/> Aprovado com ressalvas <input type="checkbox"/> Nesse caso, data limite: dd/ mm/ aaaa		dd/ mm/ aaaa		

A4.6 – MANUAL DO USUÁRIO

ID documento: **Data:** / / **Versão :**
Responsável pelo documento:

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

<Essa estrutura de Manual de Usuário segue a sugestão utilizada na FIPP – Faculdade de Informática de Presidente Prudente/UNOESTE pelos alunos que desenvolvem sistemas durante o estágio supervisionado.>

1. Introdução

<descrever o objetivo do manual do usuário>

2. Descrição do Produto

<aqui devem ser inseridas as funções e perspectivas do produto: itens 2.1 e 2.2 do Documento de Requisitos>

Perspectiva do Produto

Funções do Produto

3. Procedimentos

<Sugere-se que sejam inseridas as figuras mostrando os passos de cada um dos procedimentos, a seguir.>

Instalação do Produto

Configuração do Produto

Inicialização do Produto

Encerramento do Produto

4. Como preencher campos das telas do produto

< Sugere-se que sejam inseridas as figuras mostrando as telas que são acionadas por cada opção selecionada do sistema junto à descrição de cada um dos campos da tela que necessitam de uma explicação para ser melhor compreendido. >

5. Rotinas de Backup e Restauração

6. Mensagens de Advertência e Erro

< Para cada mensagem mostre o significado e a solução, caso seja uma mensagem de erro >

Mensagem:

Significado:

Solução:

Mensagem:

Significado:

Solução:

A4.7 – MODELO DE NEGÓCIO

ID do documento:

Data documento: / /

Responsável pelo documento:

ID Projeto:

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

Vocabulário do Negócio

< Nessa seção devem ser descritos os principais termos pertinentes ao negócio, a fim de esclarecer a terminologia utilizada pelo cliente.>

Termo	Descrição	Sinônimo

Casos de Uso de Negócio

< Um **caso de uso de negócio** é uma seqüência de ações realizada em um negócio que produz um resultado de valor observável para um ator do contexto do negócio.

Os processos de um negócio são definidos como vários casos de uso de negócio diferentes, cada um representa um fluxo de trabalho específico no negócio. Uma coleção de casos de uso de negócio fornece uma visão geral do negócio muito útil para informar aos funcionários sobre as diferentes partes do negócio que estão sendo realizadas e os resultados esperados. Um processo de negócio gera valor para o negócio ou minimiza os custos para o negócio. A figura 1 mostra um exemplo de caso de uso de negócio.

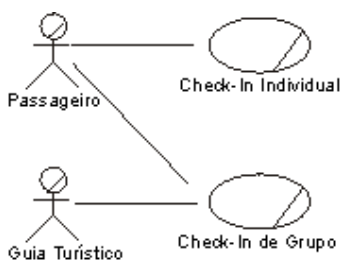


Figura 1 – Caso de uso de negócio relacionado ao Check-In no Aeroporto

Fonte: <<http://www.wthreex.com/rup/>>

Em uma modelagem de negócios controlada por caso de uso são desenvolvidas duas visões do negócio:

- **visão externa do negócio:** o **caso de uso de negócio** define “o quê” é essencial realizar para que o negócio forneça os resultados desejados ao ator. Define, também, **qual** interação o negócio deve ter com os atores quando é executado.
- **visão interna do negócio:** a **realização de casos de uso de negócio** define **como** o trabalho deve ser organizado e realizado para alcançar os resultados desejados. Uma realização abrange os trabalhadores de negócios, as entidades de negócios envolvidas na execução de um caso de uso de negócio e os relacionamentos entre eles.

Diagramas de atividades da UML e diagramas de workflow podem ser utilizados para visualizar a descrição do processo de negócios, ou seja, as atividades do negócio, o fluxo de informação entre elas e a origem e o destino das informações.

Um diagrama de atividades, por exemplo, pode ser usado para ilustrar o fluxo de trabalho de um caso de uso de negócio. Esse diagrama explora a ordem das atividades que são realizadas para alcançar as metas do negócio. Dentre essas atividades pode-se ter atividades manuais e/ou automatizadas. A figura 2 apresenta um diagrama de atividades relacionado ao caso de uso de negócio “Check-In Individual” do modelo de casos de uso de negócio Check-In no Aeroporto (figura 1).

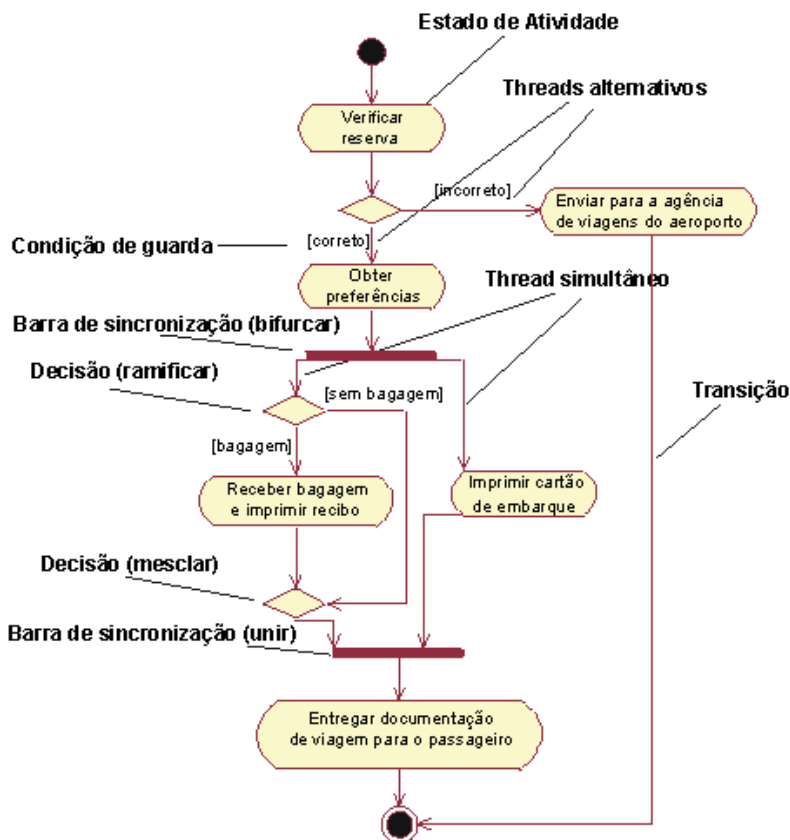


Figura 2 – Diagrama de atividades do caso de uso de negócio “Check-In Individual”

Fonte: <<http://www.wthree.com/rup/>>

A figura 3 ilustra o fluxo de trabalho de um caso de uso de negócio que representa um processo de vendas geral. Nesse exemplo, as raias representam departamentos na organização.

Referência: RUP – Rational Unified Process. Modelo integrado à ferramenta CASE Rational Software.

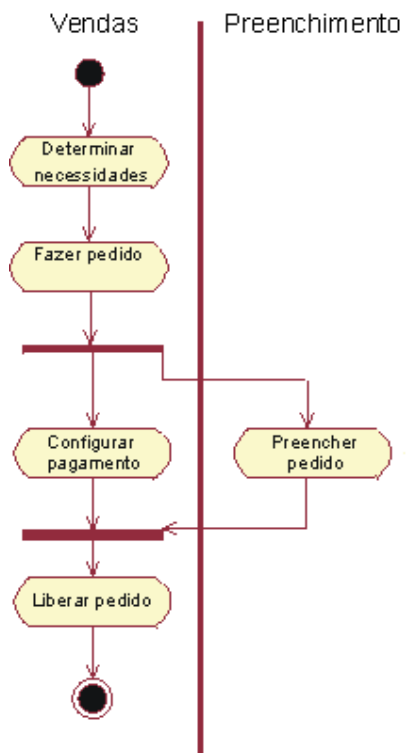


Figura 3 – Digrama de Atividades do processo de negócio de vendas.

Fonte: <<http://www.wthreex.com/rup/>>

>

Regras do Negócio

< Regras de negócio (também chamadas de regras de domínio) descrevem tipicamente requisitos ou políticas que transcendem um projeto de software – elas são necessárias no domínio ou no negócio e muitas aplicações podem precisar respeitá-las. Um exemplo são as leis sobre impostos governamentais. Os detalhes das regras de domínio podem ser registrados na especificação dos requisitos, mas como elas são usualmente mais duradouras e aplicáveis do que para um projeto de software, colocá-las em um artefato central de regras de negócio (compartilhado por todos analistas da empresa) leva ao melhor reuso do esforço de análise.

Exemplo:

Id Regra	Regra	Mutabilidade	Fonte
Regra 1	Regras de desconto para compradores. Empregado: 20% Cliente Preferencial: 10% Idoso: 15%	Alta. Cada varejista usa regras diferentes.	Política de Varejo.
Regra 2	Restituições de pagamento a crédito podem somente ser pagas como crédito à conta de crédito do comprador, não em dinheiro.	Baixa.	Política da companhia de autorização de crédito.

Referência: LARMAN, C. Utilizando UML e Padrões: Uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo. Porto Alegre: Bookman, 2007.

Para marcar quais das regras de negócio devem ser atendidas pelo software a ser desenvolvido, sugere-se uma coluna na tabela anterior que define o(s) ID(s) do(s) projeto(s) os quais as regras de negócio são utilizadas.

No exemplo, a seguir, a regra 1 é utilizada tanto no projeto “AAA” quanto no projeto “BBB” daquele cliente.

Id	Regra	Mutabilidade	Fonte	ID Projeto
Regra 1	Regras de desconto para compradores. Empregado: 20% Cliente Preferencial: 10% Idoso: 15%	Alta. Cada varejista usa regras diferentes.	Política de Varejo.	AAA BBB

Uma regra de negócio define ou restringe um aspecto do negócio, a fim de afirmar a estrutura do negócio ou influenciar o comportamento do negócio.

As regras de negócio definem como o negócio funciona, podem abranger suas políticas, interesses, objetivos, compromissos éticos e sociais, obrigações contratuais, decisões estratégicas, leis e regulamentações, entre outros.

Regras de negócio, freqüentemente, focam questões de controle de acesso, podem, também, fazer parte dos cálculos nos negócios e focam as políticas da organização. Por exemplo:

- *professores têm permissão de acessar e modificar as notas dos estudantes das disciplinas as quais ele é responsável somente;*
- *conversão de uma nota de percentual (como 91%) para uma letra significativa no contexto (como “A”);*
- *a política da universidade é reprovar no termo cursado caso o aluno reprove em mais de duas disciplinas no mesmo semestre.*
- *Professores assistentes só podem administrar grades de estudantes se autorizados pelos professores efetivos.*

>

A4.8 – MODELO DE DESIGN

ID documento: _____ **Data:** / / _____ **Versão :** _____
Responsável pelo documento: _____

ID Projeto: _____
ID do Documento de Requisitos: _____
ID da Arquitetura do Software: _____

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

1 Modelo Lógico do Sistema

< *Esse modelo tem como objetivo mostrar os elementos que irão compor o software e os relacionamentos entre eles. Pode ser representado por:*

Alternativa 1:

- **Diagrama de Classes:** contém as classes do sistema que podem estar organizadas em pacotes e subsistemas. Um pacote deve ser usado em casos onde um conjunto de classes e/ou outros pacotes necessitam ser agrupados para modelar os objetivos da organização. O conteúdo do pacote pode ter visibilidade pública. Um subsistema deve ser usado em casos onde um conjunto de classes e/ou outros pacotes necessitam ser encapsulados dentro de um container e escondidos por meio de um conjunto de interfaces bem definido. Por convenção, nenhum conteúdo dos subsistemas é visível, exceto as interfaces. Isso permite facilidade na substituição e oferece maior grau de encapsulamento em comparação aos pacotes.
- **Diagrama de Interação** para cada caso de uso, a fim de definir quem são os métodos das classes.
- **Mapeamento OO-Relacional:** como normalmente são usados bancos de dados relacionais, é necessário o mapeamento das classes para tabelas.

Alternativa 2:

- Diagrama Entidade-Relacionamento
- Diagrama de Estrutura Modular

>

2 Modelo Físico de Dados

< *Representa todas as tabelas do banco de dados a serem criadas com a definição de seus respectivos campos.*>

A4.9 – PEDIDO DE COMPRA

ID documento: _____ **Data:** / / **Versão :** _____
Responsável pelo documento: _____
ID Projeto: _____

Status do Pedido de Compra

() Congelado – Data: / /

() Liberado – Data: / /

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

1. DESCRIÇÃO DOS REQUISITOS

<Aqui são descritos os requisitos validados esperados do(s) produto(s) a ser(em) adquirido(s).>

2. DADOS DA SOLICITAÇÃO DE COMPRAS

Descrição	Quantidade	Data limite de entrega	Justificativa

A4.10 – PLANO DE GARANTIA DA QUALIDADE

ID documento: _____ Data: / / _____ Versão : _____

Responsável pelo documento: _____

ID Projeto: _____

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

1. Critérios de qualidade do projeto

< Aqui são definidos os critérios de qualidade esperados do projeto desenvolvido. Esses critérios podem incluir, além dos critérios estabelecidos pela empresa desenvolvedora (tais como: prazo de entrega, conformidade com os templates definidos no processo), os critérios de aceitação do produto determinados pelo cliente durante a definição do escopo do projeto. Alguns exemplos de critérios: usabilidade, confiabilidade, manutenibilidade, portabilidade, eficiência.>

2. Métodos de garantia da qualidade do processo

< Aqui são definidos os métodos que podem ser utilizados para garantia a qualidade do processo definido. No modelo de processo genérico, elaborado nesse trabalho de pesquisa de doutorado foi adotada a utilização da análise de conferência do processo definido durante a realização de um projeto (averiguar se o processo está realmente sendo seguido). Essa análise de conferência é realizada a cada término de fase, junto à revisão do milestone.>

3. Medidas/Métricas

Objetivos de medição

Métricas de produto

< Um exemplo de métrica de produto é: a atratividade da interface do software, ou seja, quão atrativa é a interface para o usuário? Pode ser utilizado um questionário para a obtenção dessa medida.>

Métricas de processo

< Um exemplo de métrica de processo é: a estabilidade da especificação funcional, ou seja, qual estável foi a especificação funcional durante o ciclo de desenvolvimento de software? Para obter essa medida é preciso contar o número de funções alteradas (adicionadas, modificados ou excluídas) durante o ciclo de desenvolvimento (A) e então comparar com o número de funções descritas no Documento de Requisitos (B).

Medida = $1 - A/B$ => quanto mais próximo o resultado de 1, mais estável a especificação é.>

4. Estratégia de coleta, armazenamento e análise das medidas

<No caso do Modelo de Processo Genérico, por exemplo, a medida tamanho do software é coletada para estimar o esforço e custo que pode ser despendido no projeto. A estratégia adotada para coletar essa medida é utilizando inicialmente a experiência dos especialistas que trabalham na empresa e, com o passar do tempo, por meio de uma base de dados histórica onde serão registradas medidas como tamanho do software, tempo utilizado para desenvolvimento, recursos utilizados, capacidade do pessoal, entre outras medidas, para conseguir estimar o esforço e custo de um novo projeto, a partir de projetos similares já desenvolvidos pela empresa.>

Referência Bibliográfica:

SOMMERVILLE, I. **Engenharia de Software**. 8ª ed. São Paulo: Pearson Addison Wesley, 2007.

Segundo Sommerville (2007), Garantia de Qualidade é o processo de definição de como a qualidade de software pode ser atingida e como a organização de desenvolvimento sabe que o software possui o nível de qualidade necessário. O processo de garantia de qualidade está, principalmente, relacionado à definição e seleção de padrões que devem ser aplicados ao processo de desenvolvimento de software ou ao produto de software.

Garantia da Qualidade de Software tem o objetivo de fornecer a garantia de que processos e produtos de trabalho estejam em conformidade com planos e provisões pré-definidos.

Humphrey (1989) sugere uma estrutura geral para um **plano de qualidade**. Essa estrutura inclui:

- *Apresentação do produto*. Descrição do produto, o mercado pretendido e as expectativas de qualidade para o produto.
- *Plano de produto*. As datas críticas de liberação e as responsabilidades para o produto junto com planos para serviço de distribuição e de produto.
- *Descrições de processo*. Os processos de desenvolvimento e de serviços devem ser usados para o desenvolvimento e gerenciamento de produto.
- *Metas de qualidade*. As metas e os planos de qualidade para o produto incluindo identificação e justificativa de atributos críticos de qualidade de produto.
- *Riscos e gerenciamento de riscos*. Os riscos principais que poderiam afetar a qualidade de produto e as ações para tratar esses riscos.

Em um plano de qualidade deve-se definir os atributos de qualidade mais importantes para o software em desenvolvimento. Exemplos: segurança, confiabilidade, adaptabilidade, portabilidade, facilidade de uso, facilidade de reuso, eficiência.

As revisões são métodos de validação de qualidade de um processo ou produto amplamente usados. Alguns tipos de revisão são apresentados, a seguir:

- *Inspeções de programa*: tem o objetivo de detectar erros detalhados nos requisitos, projeto ou código. Uma lista de verificação de possíveis erros deve guiar a revisão.
- *Revisões de progresso*: o objetivo é fornecer informações para a gerência sobre o progresso geral do projeto. Esta é uma revisão de processo e de produto e concentra-se em custos, planejamentos e prazos.
- *Revisões de qualidade*: o objetivo é conduzir uma análise técnica dos componentes de produto ou documentação para encontrar inconsistências

entre especificação e projeto, código ou documentação de componente e assegurar que padrões de qualidade definidos foram seguidos.

A **medição de software** se dedica a derivar um valor numérico para algum atributo de um produto de software ou de um processo de software. Alguns exemplos de medidas de produto: tamanho do software, número de defeitos. Alguns exemplos de métricas de processo: o esforço médio e o tempo necessário para reparar os defeitos reportados, o esforço total em termos de pessoas-dia, custos de viagem e recursos computacionais para um processo específico. Comparando-se esses valores uns com os outros e aos padrões que se aplicam em uma organização, você pode ser capaz de tirar conclusões sobre a qualidade de software ou dos processos de software (SOMMERVILLE, 2007). Por ex: uma organização planeja introduzir uma nova ferramenta de teste de software. Antes de introduzir a ferramenta, pode-se registrar o número de defeitos do software descobertos em um dado tempo; depois da introdução da ferramenta, pode-se concluir que ela fornece apoio útil ao processo de validação de software.

Existem duas maneiras nas quais as medições de software podem ser usadas:

- *Para fazer previsões gerais sobre um sistema.* Ao medir as características dos componentes de sistema e, em seguida, agregar essas medições, pode-se derivar uma estimativa geral de algum atributo de sistema, como o número de defeitos no sistema.
- *Para identificar componentes anômalos.* As medições podem identificar componentes individuais cujas características desviem de alguma regra. Por ex: é possível medir os componentes para descobrir os com maior complexidade e, supondo que esses são provavelmente os que terão mais erros, concentra-se sobre esses componentes durante o processo de revisão.

Para escolher as medições a serem realizadas, pode-se utilizar **a abordagem GQM (Goal-Question-Metric)** de Basili (BASILI & ROMBACH, 1988). Essa abordagem conta com a identificação de:

1. *Objetivos:* definem o que a organização está tentando obter. Exemplos: o aumento da produtividade de programadores, redução do tempo de desenvolvimento de produto, incremento da confiabilidade do produto.
2. *Questões:* São refinamentos dos objetivos. Normalmente, um objetivo terá um número de questões associadas que necessitam ser respondidas. Exemplo: Como o tempo necessário para finalizar os requisitos de produto pode ser reduzido?
3. *Métricas:* são as medições que necessitam ser coletadas para auxiliar a responder às questões e confirmar se os aprimoramentos de processos alcançaram o objetivo desejado. Exemplo de medição da questão anterior: o número de comunicações formais entre o cliente e o contratado para cada mudança de requisitos.

Segundo a ISO/IEC 9126 (*Software Engineering – Product Quality*) há três tipos de métricas:

- *Métricas internas:* podem ser aplicadas em um produto de software não-executável durante os estágios de desenvolvimento, tais como: solicitação de proposta, definição de requisitos, especificação de design ou código-fonte. Essas métricas fornecem ao usuário a habilidade de medir a qualidade das entregas intermediárias e, portanto, prediz a qualidade do produto final. Assim, é possível identificar questões de qualidade e iniciar ações corretivas tão logo quanto possível no ciclo de desenvolvimento de software.

- *Métricas externas*: podem ser usadas para medir a qualidade do produto de software por meio da medição do comportamento do sistema do qual faz parte. As métricas externas podem ser usadas durante os estágios de teste do processo de ciclo de vida e durante alguns estágios de operação. A medição é realizada quando o software está sendo executado no ambiente do sistema no qual se pretende operar.
- *Métricas de qualidade em uso*: medem se um produto atende às necessidades dos usuários para alcançar objetivos específicos com efetividade, produtividade, segurança e satisfação no contexto de uso especificado. Isso pode ser alcançado no ambiente real onde o sistema operará.

A4.11 – PLANO DE GESTÃO DE CONFIGURAÇÃO

1. Introdução

1.1 Objetivo

Este documento define um plano de gerenciamento de configuração padrão da empresa. Este plano descreve os padrões e procedimentos que devem ser usados para o gerenciamento dos itens de configuração.

1.2 Siglas, abreviações e definições

Devem ser definidos todos os termos, acrônimos e abreviações necessárias para a interpretação apropriada do Plano de Gestão de Configuração.

2. Definição dos itens de configuração

Devem ser selecionados pela Equipe de Qualidade da empresa os artefatos a serem controlados como itens de configuração do sistema a ser desenvolvido.

Normalmente, o plano de projeto, documento de requisitos, programas (código-fonte), modelos de dados e casos de teste são mantidos como itens de configuração. Entretanto, a equipe de qualidade tem a liberdade de definir quais outros artefatos, desenvolvidos em meio ao processo de desenvolvimento de software, são importantes serem colocados sob a gerência de configuração para auxiliar em uma futura manutenção do sistema (seja ela corretiva, evolutiva ou preditiva).

A seguir, são listados os itens de configuração que poderão compor a *baseline* de um projeto, organizados pelas fases do processo de venda e desenvolvimento de software. Uma *baseline* serve como uma fotografia capaz de descrever um projeto em um determinado instante de sua execução.

- **Fase Prospecção**
 - Roteiro de Prospecção
 - Modelo de Negócio
 - Plano de Ação
- **Fase Concepção**
 - Modelo de Negócio
 - Documento de Requisitos
 - Plano de Projeto
 - Projeto IHM
 - Relatório de Análise dos Requisitos
 - Arquitetura do Software
 - Cronograma
 - Relatório de Viabilidade do Projeto
 - Fluxo de Caixa
 - Plano de Ação
- **Fase Negociação**
 - Proposta Técnica
 - Proposta Comercial
 - Relatório de Viabilidade do Projeto
 - Cronograma
 - Documento de Requisitos
 - Contrato
 - Plano de Projeto
 - Cronograma
 - Documento de formalização do projeto
 - Pedido de Compra
 - Plano de Ação
 - Relatório de Acompanhamento de Projeto
- **Fase Elaboração**
 - Documento de Requisitos
 - Projeto IHM
 - Modelo de Negócio
 - Relatório de Análise dos Requisitos
 - Arquitetura do Software
 - Modelo de Design
 - Planilha de Teste
 - Plano de Projeto
 - Relatório de Acompanhamento de Projeto
 - Cronograma
 - Relatório de Viabilidade do Projeto

- **Fase Construção**
 - Arquivo de Código
 - Controle de Tarefas
 - Planilha de Teste
 - Relatório de Acompanhamento de Projeto
 - Cronograma
- **Fase Transição**
 - Planilha de Teste
 - Versão do Produto
 - Documento de Aceite
 - Manual do Usuário
 - Relatório de Acompanhamento de Projeto
 - Cronograma

3. Papéis dos responsáveis pelos procedimentos de gestão de configuração

A Equipe de Qualidade, responsável por definir o Plano de Gestão de Configuração para cada projeto a ser executado, pode ser composta pelos membros responsáveis em cada fase do processo de venda e desenvolvimento de software. Por exemplo:

- **Equipe de Vendas:** responsável pelo controle dos itens de configuração da fase Prospecção. Este papel consiste em criar novas versões de itens de configuração, criar novas *baselines* e colocar estes itens de configuração na ferramenta de controle de versões.
- **Gerente de Projeto:** responsável pela revisão, aprovação e autorização de mudança nos itens de configuração do processo de venda e desenvolvimento de software. Este papel consiste em analisar a relevância da mudança, controlar o planejamento, a implementação e os testes das mudanças nos itens de configuração, bem como a alteração de qualquer documento no projeto.

4. Estratégia de identificação dos itens de configuração

A estratégia de identificação dos itens de configuração é importante para manter a rastreabilidade de todas as informações pertinentes aos itens de configuração.

Deve ser atribuído um único nome para todos os artefatos sob controle de configuração. Este nome pode refletir o tipo do item, uma parte do sistema ao qual ele se aplica, o criador do item, etc.

É interessante que se evidencie a relação entre os itens de configuração para garantir que os artefatos relacionados possuam uma mesma raiz em seus nomes.

A identificação de um item de configuração pode ser realizada da seguinte maneira:

Nome do Item de Configuração Abreviado + “_” + ID do Projeto + “_” + V(versão) + (numeração da versão)

Sendo que:

- As letras iniciais do **nome do item de configuração abreviado** devem ser maiúsculas.
- Os identificadores dos projetos (**ID do Projeto**) devem ser únicos na empresa, ou seja, nenhum projeto poderá apresentar o mesmo identificador. Sugere-se que o identificador seja composto pela sigla do projeto composta de 3 a 5 letras em maiúsculo.
- A criação dos itens de configuração deve ocorrer no momento em que for iniciada uma atividade, de qualquer fase do processo de venda e desenvolvimento de software, responsável pelo seu desenvolvimento. Todos os itens de configuração devem ser criados na versão “V0.1”.
Todos os itens de configuração devem possuir um histórico composto pelos campos, a seguir.
 - Data de criação/atualização
 - Descrição da mudança ocorrida
 - Responsável
 - Versão do documento
 - ID. da solicitação de mudança
 - Data de início da realização da mudança
 - Data de término da realização da mudança
 - Descrição da Mudança Ocorrida

5. Política de criação de *baselines*

Baseline é uma linha de referência composta por todos os itens de configuração de um projeto. O objetivo de uma *baseline* é servir de referência para o gerenciamento de mudanças de um projeto. Ela deve ser como uma fotografia capaz de descrever um projeto em um determinado instante de sua execução.

A criação de *baselines* pode ser realizada de acordo com os marcos do projeto (*milestones*) ou de alguma outra forma definida pela gerência. A *baseline* é armazenada em um repositório de itens de configuração. E, a partir desse momento, só pode ser alterado por meio de uma solicitação de alteração formal para controle de mudança.

No modelo de processo genérico elaborado nessa pesquisa de doutorado, sugere-se que quando é realizada tanto uma mudança simples em um ou mais itens de configuração quanto complexa deve ser criada uma nova *baseline* que incorpore os itens de configuração modificados, autorizada pelo Gerente de Projeto.

A fim de se ter um controle de *baselines* a cada término de fase do modelo de processo genérico, mesmo que não haja modificações, para cada uma das fases do modelo sugere-se a criação de *baselines* nos seguintes marcos:

- Fase Prospecção
 - Milestone: Solicitação de uma proposta pelo cliente
- Fase Concepção
 - Após a atividade do ciclo de desenvolvimento: Cp14. Conduzir revisão do *milestone*: Estudo de Viabilidade do Projeto
- Fase Negociação
 - Após a atividade do ciclo de desenvolvimento: N09. Conduzir revisão do *milestone*: Contrato
- Fase Elaboração
 - Após a atividade do ciclo de desenvolvimento: E08. Conduzir revisão do *milestone*: Arquitetura do Software
- Fase Construção
 - Após a atividade do ciclo de desenvolvimento: Ct05. Conduzir revisão do *milestone*: Capacidade Operacional
- Fase Transição
 - Após a atividade do ciclo de desenvolvimento: T06. Conduzir revisão do *milestone*: Entrega do Produto

A identificação da baseline pode ser constituída por:

“Baseline” + “_” + **ID do Projeto** + “_” + (numeração sequencial iniciada por 0001)

Sendo que:

- Os identificadores dos projetos (**ID do Projeto**) devem ser únicos na empresa, ou seja, nenhum projeto poderá apresentar o mesmo identificador. Sugere-se que o identificador seja composto pela sigla do projeto composta de 3 a 5 letras em maiúsculo.

6. Políticas de gestão de configuração

Nesta seção são definidas as políticas de gerenciamento de configuração que todos os membros da equipe do projeto devem adotar para o controle de mudanças e gerenciamento das versões.

A seguir, é mostrado um exemplo de políticas de gestão de configuração. Essas políticas estão inseridas no modelo de processo genérico. Caso queira modificá-las é preciso modificar as tarefas das atividades correspondentes no modelo.

Exemplo de políticas:

Os itens de configuração poderão ser modificados perante as seguintes situações:

- Enquanto o item de configuração não tiver uma primeira versão finalizada, determinada por um marco de projeto (*milestone*), ele será modificado sem alteração de versão.

- Após o término de uma versão do item de configuração, ele só poderá ser alterado para outra versão mediante uma solicitação de mudança e análise da mudança para detectar a relevância e complexidade da mesma. Nesse caso, pode haver duas situações:
 - Mudança simples: Deve ser realizada sem a necessidade de todo o processo de análise e planejamento da mudança. Devem ser realizados testes depois de finalizada a implementação da mudança. A realização de uma mudança simples deve ser registrada no próprio histórico do item de configuração que foi alterado.
 - Mudança complexa: Caso uma mudança seja considerada complexa, deve-se então seguir todo o fluxo do processo de gestão de configuração: GCf04. Analisar/Autorizar e Planejar mudança(s), CGf05. Implementar mudança(s), CGf06. Validar e Liberar mudança(s). Uma mudança complexa também deve ser registrada no histórico do item de configuração.
- A identificação da solicitação de mudança pode ser constituída por:
“*Mudança*” + “_” + *ID do Projeto* + “_” + (*numeração sequencial iniciada por 0001*)
Sendo que:
 - Os identificadores dos projetos (**ID do Projeto**) devem ser únicos na empresa, ou seja, nenhum projeto poderá apresentar o mesmo identificador. Sugere-se que o identificador seja composto pela sigla do projeto composta de 3 a 5 letras em maiúsculo.

Quando realizada tanto uma mudança simples quanto uma complexa deve ser criada uma nova *baseline* que incorpore os itens de configuração modificados, de acordo com a política de criação de *baselines* descrita na seção 5.

Um *release* do sistema é uma versão distribuída ao cliente. Cada *release* deve incorporar novas funcionalidades ou ser planejado para uma plataforma diferente de hardware. Há, normalmente, muito mais versões de um sistema do que liberações. As versões são criadas no âmbito da organização, para desenvolvimentos ou testes internos, e não são previstas para serem liberadas para os clientes.

Um *release* do sistema não é somente um código executável do sistema. Ele pode incluir: arquivos de configuração que definem como o *release* pode ser configurado para instalações específicas; arquivo de dados necessário para a operação do sistema com sucesso; um programa de instalação usado para auxiliar a instalação do sistema no hardware-alvo; documentação eletrônica e em papel que descreve o sistema; empacotamento e publicidade associada projetados para *release*.

Para documentar um *release*, deve-se:

- registrar as versões específicas dos componentes de código-fonte usados para criar o código executável;
- manter cópias dos códigos-fonte, do código-executável, de todos os arquivos de dados e de configuração;
- registrar as versões do sistema operacional, as bibliotecas, os compiladores e outras ferramentas usadas para construir o software.

7. Ferramentas utilizadas

< aqui são descritas as ferramentas utilizadas para realizar o controle de mudanças e versões dos artefatos do projeto >

8. Estrutura do Banco de Dados de Configuração

O banco de dados de configuração é utilizado para registrar todas as informações relevantes sobre as configurações de sistema e os itens de configuração. Assim, ele auxilia a avaliação do impacto das mudanças de sistema e permite gerar relatórios para a gerência.

Referência Bibliográfica: SOMMERVILLE, I. **Engenharia de Software**. 8ª ed. São Paulo: Pearson Addison Wesley, 2007.

A4.12 – PLANO DE MARKETING

Título: Deve descrever o que o plano tem como objetivo

Data Inicial: Data de início do plano

Data Final: Data final do plano

HISTÓRICO DE REVISÕES

Data de criação/atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

1. Contextualização

< É realizada a análise SWOT - trata-se da avaliação global das forças, fraquezas, oportunidades e ameaças (strengths, weaknesses, opportunities, threats). Em geral, uma empresa tem que monitorar importantes forças macroambientais (econômico-demográficas, tecnológicas, político-legais e socioculturais) e significativos agentes microambientais (clientes, concorrentes, distribuidores, fornecedores) que afetam sua capacidade de obter lucros. É realizada uma análise de mercado, considerando oportunidades e ameaças. Além disso, é feita uma análise dos pontos fortes e fracos da sua empresa e são considerados, também, os concorrentes.

A lista, a seguir, sugere itens a serem considerados na análise de forças/fraquezas:

- *Marketing: reputação da empresa, participação de mercado, satisfação do cliente, retenção do cliente, qualidade do produto, qualidade de serviço, efetividade na determinação de preços, efetividade na distribuição, efetividade de promoções, efetividade da força de vendas, efetividade das inovações, cobertura geográfica.*
- *Finanças: custo ou disponibilidade de capital, fluxo de caixa, estabilidade financeira.*
- *Produção: instalações, economias de escala, capacidade, força de trabalho capaz e dedicada, capacidade de produzir no prazo, habilidades técnicas de fabricação.*
- *Organização: liderança visionária e capaz, funcionários dedicados, orientação empreendedora, flexibilidade ou boa capacidade de resposta.*

Exemplo da Análise SWOT (forças, fraquezas, oportunidades e ameaças).

Potencialidades (Fatores Internos) 1- Fácil Implantação 2- Satisfação dos atuais clientes	Fragilidades (Fatores Internos) 1- Produto pouco conhecido pelo mercado 2- Preço acima da capacidade de grande parte do mercado
Oportunidades (Fatores Externos) 1- Baixa Concorrência 2- Existência de recursos de órgãos de fomento para aquisição do sistema	Ameaças (Fatores Externos) 1- Produto concorrente sem custo para clientes

O produto deve ser contextualizado (aproximadamente um parágrafo) e pode ser inserida a lista de atuais clientes atendidos e casos de sucesso. Além disso, é importante definir o público-alvo desse produto.

São, também, definidos o preço do produto e as condições de fornecimento do produto (no caso de ser algum produto já pronto da empresa, sem customizações).

>

2. Objetivos

<Define as metas financeiras e de marketing do plano em relação ao volume de vendas, participação do mercado e lucros e índices de satisfação dos clientes.

Exemplos de objetivos quantitativos: aumentar vendas em 10%; aumentar participação de mercado para 30%.

Exemplos de objetivos qualitativos: treinar equipe de vendas; implantar filosofia de qualidade; melhorar o relacionamento com os fornecedores.>

3. Operacionalização

3.1 Perfil de potenciais clientes

< Aqui deve ser descrito o perfil dos potenciais clientes.>

3.2 Estratégia para busca de potenciais clientes

<Aqui deve ser definido como buscar os potenciais clientes, conforme o perfil descrito no item anterior.

Exemplo:

Etapa 1 – E-mail marketing e/ou material marketing via correio

Etapa 2 – Contato Telefônico

Etapa 3 – Visita

Etapa 4 – Proposta

Etapa 5 - Fechamento

>

4. Plano de Ação

< O que será feito para atingir aos objetivos do plano? Como? Quando será feito? Quem fará? Quanto custará?>

Aqui deverá ser definido um conjunto de etapas que coloque em prática a estratégia definida anteriormente, assim como cada uma dessas etapas deverá ser realizada (seqüência e ordem de prioridade). Quem será o responsável pela execução e qual o custo estimado. Para cada etapa desse plano deverá ser definido um conjunto de informações que deverão ser transmitidas aos potenciais clientes e outro conjunto de respostas que a empresa espera para que a próxima etapa possa ser iniciada.

Exemplo: As seguintes etapas serão abordadas nas fases prospecção, concepção e negociação:

Etapa 1 – E-mail marketing

Etapa 2 – Contato Telefônico

Etapa 3 – Visita

Etapa 4 – Proposta

Etapa 5 - Fechamento

Para cada etapa deverá ser definida a forma de abordagem. Ex: etapa 1 email marketing. Definir texto do email, layout e links. Definir o que é esperado como retorno dessa ação. Ex. telefonema, email de resposta manifestando interesse ou click no link X. >

5. Atividades de inicialização e infra-estrutura

<Aqui devem ser definidas as atividades necessárias para que seja iniciada a ação de vendas. Ex: Instalação de um novo ramal de telefone na sala Y.>

6. Equipe

<Aqui deverá ser definida a equipe que será responsável pelas ações de venda, percentual de alocação de cada recurso e horário de trabalho.>

7. Metas de venda

< Aqui deverá ser definida a meta de vendas geral e parcial por período e por pessoa.>

8. Custos referentes à ação de venda

< Aqui deverá ser descrito o orçamento relativo ao planejamento da ação de venda em questão. Pessoal, telefone, infra-estrutura, viagens, comissões, eventos, material de divulgação. Para que esse item seja feito com um mínimo de confiabilidade é necessário que o item 3 deste documento (Plano de ação) tenha um planejamento não só de como cada etapa deverá ser realizada, mas também o número de eventos por etapa por período. Ex: arquivo excell plano_de_acao.xls.>

9. Acompanhamento do Plano de Ação

<Aqui deverá ser definida a forma de acompanhamento do plano. Isso deverá incluir periodicidade e itens para avaliação. Para que esse item seja realizado com sucesso é necessário que o item 3 (Definir Plano de ação) tenha um planejamento não só de como cada etapa deverá ser realizada, mas também o número de eventos por etapa e por período para que possa ser produzido uma planilha mostrando a comparação entre o planejado e o realizado.Ex: guia “Acompanhamento” do template “Plano de Ação.xls”.>

Referências:

KOTLER, P. Administração de Marketing. 10 ed. São Paulo: Pearson Prentice Hall, 2000.

LAS CASAS, A. L. Plano de marketing para micro e pequena empresa. São Paulo: Atlas, 2001.

ROSA, C. A. Como elaborar um plano de negócio. Brasília: SEBRAE, 2007.

GOMES, I. M. Como elaborar um plano de marketing. Belo Horizonte: SEBRAE/MG, 2005.

CARVALHAIS, R. S.; PATTO, A. R. Como elaborar um plano de vendas. Belo Horizonte: SEBRAE/MG, 2007.

O **planejamento estratégico** é o processo gerencial de desenvolver e manter uma direção estratégica que alinhe as metas e os recursos da organização com suas mutantes oportunidades de mercado (KOTLER, 2000).

Segundo Kotler (2000), o **planejamento estratégico de negócios** consiste de oito etapas. São elas:

- **Missão do Negócio:** cada unidade de negócio precisa definir sua missão específica dentro da missão corporativa.
- **Análise SWOT:** trata-se da avaliação global das forças, fraquezas, oportunidades e ameaças (strengths, weaknesses, opportunities, threats). Em geral, uma unidade de negócios tem que monitorar importantes forças macroambientais (econômico-demográficas, tecnológicas, político-legais e socioculturais) e significativos agentes microambientais (clientes, concorrentes, distribuidores, fornecedores) que afetam

sua capacidade de obter lucros. A unidade de negócios deve estabelecer um sistema de inteligência de marketing para acompanhar tendências e mudanças importantes. A administração precisa identificar as oportunidades e ameaças associadas a cada tendência ou desenvolvimento. Um objetivo importante da avaliação ambiental é o reconhecimento de novas oportunidades de marketing. Uma oportunidade de marketing existe quando a empresa pode lucrar ao atender às necessidades dos consumidores de um determinado segmento. Cada unidade de negócio, também precisa avaliar periodicamente suas forças e fraquezas internas. Isso pode ser feito pela gerência (ou consultor externo) que analisa as competências de marketing, financeiras, de fabricação e organizacionais. A lista, a seguir, sugere itens a serem considerados na análise de forças/fraquezas:

- *Marketing*: reputação da empresa, participação de mercado, satisfação do cliente, retenção do cliente, qualidade do produto, qualidade de serviço, efetividade na determinação de preços, efetividade na distribuição, efetividade de promoções, efetividade da força de vendas, efetividade das inovações, cobertura geográfica.
 - *Finanças*: custo ou disponibilidade de capital, fluxo de caixa, estabilidade financeira.
 - *Produção*: instalações, economias de escala, capacidade, força de trabalho capaz e dedicada, capacidade de produzir no prazo, habilidades técnicas de fabricação.
 - *Organização*: liderança visionária e capaz, funcionários dedicados, orientação empreendedora, flexibilidade ou boa capacidade de resposta.
- **Formulação de metas**: a empresa deve desenvolver metas específicas para o período de planejamento. As metas indicam aquilo que uma unidade de negócios deve alcançar. Essas metas são utilizadas para descrever objetivos em termos de magnitude e prazo.
 - **Formulação de estratégias**: a estratégia é o meio para atingir os fins (os objetivos da empresa); refere-se a como devem ser alcançadas as metas. Exemplos de estratégias: liderança total em custos (a empresa se esforça para conseguir os menores custos de produção e de distribuição, de modo a poder oferecer preços mais baixos do que os dos concorrentes e a obter uma grande participação de mercado).
 - **Formulação de programas**: devem ser definidos planos de programas baseados na estratégia definida.
 - **Implementação**: os programas devem ser executados.
 - **Feedback e Controle**: à medida que se implementa a estratégia, deve-se realizar o acompanhamento dos resultados e monitoramento dos novos acontecimentos nos ambientes interno e externo.

O plano estratégico é composto pelo plano de marketing, plano de produção, plano financeiro e plano de recursos humanos (LAS CASAS, 2001). O plano de marketing, por sua vez, é composto pelo plano de vendas, plano de propaganda, plano de novos produtos e plano de *merchandising*.

O processo de marketing consiste em analisar oportunidades de marketing, pesquisando e selecionando mercados-alvo, delineando estratégias, planejando programas e organizando, implementando e controlando o esforço de marketing.

Para cada nível de produto (linha de produtos, marca) deve ser desenvolvido um plano de marketing para atingir suas metas. Esse plano é um dos produtos do processo de marketing. Um **Plano de Marketing** contém:

- **Resumo executivo e sumário**: apresenta uma rápida visão geral do plano proposto.

- **Situação atual de marketing:** apresenta antecedentes relevantes sobre vendas, custos, lucros, mercado, concorrentes, distribuição e macroambiente.
- **Análise de oportunidades e questões:** identifica as principais oportunidades/ameaças, forças/fraquezas e questões relacionadas à linha de produtos.
- **Objetivos:** define as metas financeiras e de marketing do plano em relação ao volume de vendas, participação do mercado e lucros.
- **Estratégia de marketing:** apresenta a abordagem geral de marketing que será utilizada para alcançar os objetivos do plano. O gerente de produto, ao desenvolver a estratégia de marketing, conversa com o pessoal de compras e de fabricação, a fim de confirmar se são capazes de comprar material suficiente e produzir unidades suficientes para atender aos níveis-alvo de volume de vendas; também conversa com o gerente de vendas, para obter apoio da força de vendas, e com o gerente financeiro, para conseguir os recursos necessários para propaganda e promoção.
- **Programas de ação:** apresenta os programas especiais de marketing projetados para atingir aos objetivos do plano. O que será feito? Quando será feito? Quem fará? Quanto custará?
- **Demonstrativo de resultados projetados:** projeta os resultados financeiros esperados do plano. Os planos de ação permitem que o gerente de produto desenvolva um orçamento de apoio. Pelo lado da receita, esse orçamento mostra o volume esperado de vendas em unidades e seu preço médio. Pelo lado de despesas, mostra os custos de produção, de distribuição e de marketing. Uma vez aprovado, o orçamento é a base para desenvolver planos e programações de suprimento de materiais, de produção, de recrutamento de funcionários e de operações de marketing.
- **Controles:** indica como o plano será monitorado. As metas e orçamentos foram especificados detalhadamente para cada mês ou trimestre. A alta administração pode analisar os resultados a cada período. Algumas seções de controle incluem planos de contingência. Um plano de contingência descreve as atitudes que a gerência tomaria em resposta a eventos adversos específicos, como guerras de preço.

Um roteiro simplificado para um plano de marketing é apresentado por Las Casas (2001):

1. Análise Ambiental

- a. Ameaças e oportunidades: análise do mercado como um todo.
Para cada evento, descrever ameaças, oportunidades e sugestões.
- b. Pontos fortes e fracos: realizada com o propósito de identificar quais empresas concorrentes têm melhores condições de aproveitar as oportunidades ou se defender de eventuais ameaças.
Os aspectos que podem ser analisados são: pessoal (quantidade e qualificação), equipamentos (capacidade instalada, tecnologia), finanças (recursos financeiros, possibilidade de obtenção de empréstimos), marketing (produto, preço, distribuição, propaganda, equipe de vendas, promoção).

2. Objetivos:

- a. Quantitativos
Exemplos: aumentar vendas em 10%; aumentar participação de mercado para 30%.
- b. Qualitativos:

Exemplos: treinar equipe de vendas; implantar filosofia de qualidade; melhorar o relacionamento com os fornecedores.

3. Estratégia de Marketing

a. Público-alvo

Exemplo: segmento de homens, faixa etária de 25 a 35 anos, renda entre R\$ 1.500,00 e R\$ 3.000,00.

b. Posicionamento

Exemplo: produto de boa qualidade e preço intermediário.

c. Estratégia do mix: produto, preço, distribuição e promoção.

Exemplo: o produto a ser comercializado será o relógio Champion, com qualidade suíça, modelo quartzo, linha esportiva, plástico resistente, à prova d'água e com pulseiras intercambiáveis de várias cores. A marca é Champion; o produto terá garantia de 4 anos; seu preço é de R\$ 50,00; será distribuído por varejistas, principalmente relojarias e lojas de departamentos. Quanto à propaganda, será utilizado um outdoor no lançamento, filmes de 15" na TV e anúncios nas revistas. Quanto à promoção de vendas: promoção junto aos varejistas com uso de displays e cartazes; relações públicas: assessoria de imprensa.

4. Plano de Ação: para cada atividade, descrever encarregado, período e orçamento.

Exemplo:

<i>Atividades</i>	<i>Encarregado</i>	<i>Período</i>	<i>Orçamento</i>
<i>Promoção: uso de outdoor no lançamento</i>	<i>José Silva</i>	<i>Jan a mar de 2008</i>	<i>(incluir valores estimados)</i>
<i>Filmes de 15" na TV: Globo e Bandeirantes</i>	<i>José Silva</i>	<i>Jan a jul de 2008</i>	<i>(incluir valores estimados)</i>
<i>Anúncios em revistas: Veja e Exame</i>	<i>Maria Silva</i>	<i>Mar de 2008</i>	<i>(incluir valores estimados)</i>

5. Projeção de vendas e lucros: para cada período, estabelecer vendas e lucros e fazer observações pertinentes.

A4.13 – PLANO DE PROJETO

ID documento: **Data:** / / **Versão :**

Responsável pelo documento:

ID Projeto:

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

1. ESCOPO DO PROJETO

1.1 Cenário

< Identificar as necessidades que motivaram o cliente a solicitar o projeto, definir os stakeholders e usuários do sistema (quem são e quais as necessidades de cada um) e descrever as expectativas que o projeto visa atender >

1.2 Entregas do Projeto

<As entregas são desdobramentos do produto de projeto em resultados tangíveis. Podem ser físicas ou uma mudança de estado visível. A característica fundamental das entregas é a capacidade de serem medidas e avaliadas. Por ex: documento de requisitos, relatório de teste, etc.>

1.3 Premissas, Limitações e Restrições do Projeto

1.4 Critérios de aceitação do cliente

< Descrever como o projeto atenderá às expectativas do cliente e itens de sucesso para o cliente, tal como antecipação de prazos.>

1.5 Estratégias

1.6 Prazos Máximos

1.7 Custo e Preço-Meta

2. PLANO DE RECURSOS

<Aqui devem ser descritos todos os recursos (equipamentos físicos, tais como hardware, impressora e outros periféricos) necessários para o desenvolvimento do software.>

3. PLANO DE HABILIDADES E CONHECIMENTO

<Aqui devem ser descritos todos os conhecimentos e habilidades necessários para o desenvolvimento do software e na frente de cada um deles, sugerir colaboradores que possam participar como integrantes do projeto>

4. RISCOS

Tabela de Riscos						
Descrição	Escala	Probabilidade	Impacto	Prioridade	Ações	% Risco do projeto

< Lista de possíveis riscos: Horas extras; Rotatividade/perda de colaboradores; Necessidade de equipamentos; Falta de colaboração do cliente; Falta de conhecimento do domínio >

5. PLANO DE TESTES

Referência bibliográfica: [Introdução ao Teste de Software](#)

5.1 Critérios de Teste de Unidade

5.2 Critérios de Teste de Integração

5.3 Teste de Validação

Tempo máximo:

5.4 Ferramentas de Teste

5.5 Equipamentos Necessários

APÊNDICE 1: PLANO DE CONTINGÊNCIA

<inserir aqui o plano de contingência>

ANEXO 1: CRONOGRAMA

ID:

ANEXO 2: FLUXO DE CAIXA (PARA MONITORAMENTO)

ID:

ANEXO 3: RELATÓRIO DE VIABILIDADE DO PROJETO

ID:

ANEXO 4: DOCUMENTO DE REQUISITOS

ID:

ANEXO 5: PLANILHA DE TESTE

A4.14 – PROJETO INTERFACE HOMEM-MÁQUINA

ID documento: _____ Data: / / _____ Versão : _____
 Responsável pelo documento: _____

ID Projeto: _____
 Documento de requisitos: _____

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

1 Perfil do Usuário

< Para cada tipo de usuário previsto, os projetistas devem conhecer seus atributos pessoais (faixa etária, sexo, limitações, motivação) e suas habilidades e competências (na tarefa, na organização e com sistemas informatizados).

Tipo de Usuário	Faixa etária	Sexo	Limitações	Motivação	Habilidades	Competências

>

2 Princípios Gerais para o Projeto

< Pesquisa e catalogação do conhecimento ergonômico disponível para a concepção da interface no tipo de contexto de uso (usuário, tarefa, equipamento e ambiente) no qual o sistema está inserido. Podem ser utilizados como referência para a pesquisa e catalogação do conhecimento ergonômico, os critérios ergonômicos propostos por Scapin & Bastien (1993). Esses critérios são apresentados no Apêndice 1.>

3 Padrão de Telas

<São definidas regras para a escolha de controles, para a definição do formato e localização das telas, para a terminologia empregada, para o uso de cores, tipos de fontes, etc.

Deve-se construir o layout das telas com base nessas regras. Inclusive, pode-se construir uma maquete informatizada (considerada um protótipo executável com baixa fidelidade) de uma parte da interface, de modo que seja possível um diálogo com o usuário, sem uma base de dados. No que se refere à modelagem de websites, a ferramenta Denim é uma alternativa e pode ser obtida gratuitamente no site <http://dub.washington.edu/projects/denim/download>. Na falta de uma ferramenta especializada, é sempre possível usar um editor de apresentações, como o PowerPoint. >

4 Mapa de Navegação

<Para representar o mapa de navegação pode ser utilizado o diagrama de transição de estados, no qual os espaços de interação são representados por retângulos, e as transições são representadas por flechas conectando espaços.>

Referência: CYBIS, W.; BETIOL, A. H.; FAUST, R. Ergonomia e Usabilidade: Conhecimentos, Métodos e Aplicações. São Paulo: Novatec Editora, 2007.

Ciclo de engenharia de usabilidade

Este modelo, proposto por Mayhew (1999), tem a mesma estrutura do modelo proposto pela norma ISO 13407. Entretanto, o modelo de Mayhew fornece mais detalhes sobre o conteúdo das atividades a serem realizadas em cada fase do processo de desenvolvimento.

- 1) **Análise de Requisitos:** os resultados das atividades dessa fase são utilizados para especificar o contexto de uso e a usabilidade pretendida para o sistema. As atividades dessa fase são:
 - **Análise do perfil do usuário:** para cada tipo de usuário previsto, os projetistas devem conhecer seus atributos pessoais (faixa etária, sexo, limitações, motivação) e suas habilidades e competências (na tarefa, na organização e com sistemas informatizados).
 - **Análise do contexto da tarefa:** para cada tarefa a ser apoiada pelo sistema, os projetistas devem conhecer os objetivos e resultados, a estrutura, a duração, as dependências, os custos, a carga mental, as interrupções, os incidentes, etc.
 - **Análise das possibilidades e restrições da plataforma:** são examinadas as possibilidades e restrições em termos de equipamentos, sistemas operacionais, ambientes de janelas, recursos de rede, a possibilidade de oferecer recursos em termos de manuais, suporte, treinamento, etc.
 - **Análise de princípios gerais para o projeto:** atividade de pesquisa e catalogação do conhecimento ergonômico disponível para a concepção da interface no tipo de contexto de uso (usuário, tarefa, equipamento e ambiente) no qual o sistema está inserido. Podem ser utilizados como referência para a pesquisa e catalogação do conhecimento ergonômico, os oito critérios ergonômicos propostos por Scapin & Bastien (1993). Esses critérios são apresentados no Apêndice 1.
 - **Especificação do contexto de uso:** o projetista especifica que tipo de usuário irá operar o sistema para realizar que tipo de tarefa e em

que condições ambientais (qual software, equipamento, ambiente físico e organizacional).

- **Especificação das exigências para a usabilidade:** são especificadas as exigências qualitativas para a interface e quantitativas para a usabilidade.
 - i. As exigências qualitativas referem-se às funções e características da interface de modo a satisfazer o tipo de usuário. Por exemplo, se a maioria são idosos, então uma função de zoom deveria ser implementada e esses usuários deveriam fazer parte da população de testes do sistema.
 - ii. As exigências quantitativas referem-se ao nível de usabilidade esperado para o sistema. Essa especificação é realizada em termos de valores mínimos admissíveis para os fatores básicos de usabilidade: eficácia, eficiência e satisfação do usuário principalmente. Exemplo:

Objetivos da usabilidade	Medidas de eficácia	Medidas de eficiência	Medidas de satisfação
Usabilidade global	% de objetivos alcançados	Tempo para completar a tarefa	Escala de satisfação
	% de usuários completando a tarefa com sucesso	Tarefas completadas por unidade de tempo	Frequência de uso
	Média da acurácia de tarefas completadas	Custo monetário de realização da tarefa	Frequência de reclamações

- 2) **Projeto, Testes e Implementação:** os sucessivos ciclos dessas fases envolvem três versões de uma mesma interface: modelo conceitual da interface, padrão de telas e projeto detalhado da interface.
 - Modelo Conceitual da Interface: esse modelo pode ser definido como representações abstratas de alternativas de projeto, nas quais são especificadas as principais telas e componentes da interface, bem como a navegação entre elas. Para representar o modelo conceitual pode-se usar uma maquete (protótipo em papel). Para representar o mapa de navegação pode ser utilizado o diagrama de transição de estados, no qual os espaços de interação são representados por retângulos, e as transições são representadas por flechas conectando espaços.
 - Padrão de Telas: são definidas regras para a escolha de controles, para a definição do formato e localização das telas, para a terminologia empregada, para o uso de cores, tipos de fontes, etc. Pode-se construir uma maquete informatizada (considerada um protótipo executável com baixa fidelidade) de uma parte da interface, de modo que seja possível um diálogo com o usuário, sem uma base de dados. No que se refere à modelagem de websites, a ferramenta Denim é uma alternativa e pode ser obtida gratuitamente no site <http://dub.washington.edu/projects/denim/download>. Na falta de uma ferramenta especializada, é sempre possível usar um editor de apresentações, como o PowerPoint.
 - Projeto Detalhado da Interface: os aspectos definidos no modelo conceitual e no padrão de telas são integrados.
- 3) **Instalação:** depois que o usuário já está acostumado em utilizar o software, o seu *feedback* é valioso para: detectar e eliminar problemas e preparar um novo

release do produto; detectar oportunidades para melhoria de novas versões do produto e elaborar requisitos para novos produtos similares. Para coletar esse *feedback* são utilizados testes de usabilidade no local de trabalho dos usuários ou métodos de análise, tais como observações, entrevistas e questionários.

As técnicas de avaliação, a seguir apresentadas, têm como foco duas qualidades: a ergonomia das interfaces e a usabilidade dos sistemas. A ergonomia é a qualidade da adaptação de um dispositivo a seu operador e à tarefa que este realiza. A usabilidade se revela quando usuários empregam o sistema para alcançar seus objetivos em um determinado contexto de operação, sendo caracterizada pelo nível de eficácia, eficiência e satisfação alcançado pelo usuário durante o seu uso (conforme ISO 9241:11). Algumas alternativas de questionário de satisfação podem ser encontradas em http://www.usabilitynet.org/tools/r_questionnaire.htm, tais como: SUMI – Software Usability Measurement Inventory (5 fatores; 50 questões) <http://www.ucc.ie/hfrg/questionnaires/sumi/index.html>; SUS – System Usability Scale (10 questões). Uma alternativa interessante é a versão em português do questionário ISONORM, desenvolvida por Medeiros (1999), <http://teses.eps.ufsc.br/defesa/pdf/1073.pdf>.

As inspeções de ergonomia, por meio de listas de verificação, permitem que profissionais, não necessariamente especialistas em ergonomia, identifiquem problemas menores e repetitivos das interfaces. As normas ISO 9241, partes 10 a 17, fornecem listas de verificação de ergonomia bem-definidas, assim como as listas de verificação fornecidas pelo site ErgoList (<http://www.labiutil.inf.ufsc.br/ergolist>).

Os testes de usabilidade têm como foco de avaliação a qualidade das interações que se estabelecem entre usuários e o sistema. Um teste de usabilidade envolve usuários reais ou representativos da população-alvo do sistema interagindo com ele para realizar tarefas específicas em um contexto de operação real ou simulado.

APÊNDICE 1 - Critérios Ergonômicos segundo Scapin & Bastien (1993)

- o **Condução**: a interface deve aconselhar, orientar, informar e conduzir o usuário na interação com o sistema. Essa qualidade pode ser analisada a partir de quatro sub-critérios:
 - **Convite**: engloba os meios utilizados para levar o usuário a realizar determinadas ações. Uma interface convidativa apresentará: títulos claros para as tarefas, janelas e caixas de diálogo; informações claras sobre o estado (disponível, em foco selecionado, etc) dos componentes do sistema; informações sobre o preenchimento de um formulário, sobre as entradas esperadas; opções de ajuda claramente indicadas.
 - **Agrupamento/Distinção de itens**: a rápida compreensão de uma tela pelo usuário depende, dentre outros itens, do posicionamento, da ordenação e da forma dos objetos (imagens, textos, comandos, etc) apresentados. Esse critério é composto pelos seguintes sub-critérios: agrupamento/distinção por localização (quando o usuário percebe rapidamente os grupamentos a partir da localização das informações nas interfaces) e agrupamento/distinção por formato (quando o usuário percebe rapidamente as similaridades ou diferenças entre as informações, a partir da forma gráfica de componentes da interface, como tamanho, cor da figura e do fundo, estilo dos caracteres, etc).

- **Legibilidade:** diz respeito às características que possam dificultar ou facilitar a leitura das informações textuais. Deve ser considerada, principalmente, quando os usuários são pessoas idosas ou com problemas de visão (baixa visão). Em uma interface legível: o texto longo que deve ser lido rapidamente aparece em letras maiúsculas e minúsculas naturalmente ao invés de somente com maiúsculas; o texto é apresentado em linhas com comprimento adequado e com um contraste efetivo com o fundo; o texto que deve ser lido por idosos ou pessoas com problemas de visão aparece em letras claras sobre um fundo escuro. Para essas pessoas, o fundo brilhante pode ofuscar completamente as letras escuras.
- **Feedback imediato:** está a serviço de todos, porém os mais novatos precisarão mais dessa qualidade. Uma interface que fornece feedback de qualidade: relata ao usuário o recebimento de todas as entradas por ele efetuadas (as entradas confidenciais são relatadas de modo a não revelar o seu conteúdo, por exemplo, com asteriscos); indica ao usuário que um tratamento demorado está sendo realizado, bem como a sua conclusão e o seu resultado.
- o **Carga de trabalho:** esse critério se aplica, principalmente, a um contexto de trabalho intenso e repetitivo, no qual os profissionais que operam o sistema precisarão de interfaces econômicas sob o ponto de vista cognitivo e motor, ou seja, que lhes economizem leitura e memorização desnecessárias, assim como deslocamentos inúteis e repetição de entradas. Os sub-critérios desse são:
 - **Brevidade:** as qualidades inerentes a esse são a concisão e as ações mínimas. Uma **interface concisa** apresenta títulos (de telas, janelas e caixas de diálogo), rótulos (de campos, de botões e de comandos) e denominações curtas; apresenta códigos arbitrários (nome de usuário, senha) curtos; fornece valores default (para os campos de dados, lista, *check boxes*) capazes de acelerar as entradas individuais e fornece o preenchimento automático de vírgulas, pontos decimais e zeros à direita da vírgula nos campos de dados. Uma **interface ágil** não solicita aos usuários dados que podem ser deduzidos pelo sistema; não força o usuário a percorrer em seqüência todas as páginas de um documento de modo a alcançar a página específica e não solicita o mesmo dado ao usuário diversas vezes em uma mesma seqüência de diálogo.
 - **Densidade informacional:** critério a serviço principalmente de usuários iniciantes, os quais podem encontrar dificuldades para filtrar a informação de que necessitam em uma tela carregada. Uma interface minimalista apresenta somente os itens que estão relacionados à tarefa (o restante deve ser removido da tela); não força os usuários a transportar mentalmente dados de uma tela a outra; não força os usuários a realizar procedimentos complicados, como a transformação da unidade de medida; não coloca os usuários diante de tarefas cognitivas complexas, como as de especificação de buscas avançadas.
- o **Controle explícito:** aplica-se em particular às tarefas longas seqüenciais e nas quais os processamentos são demorados. Seus sub-critérios são:
 - **Ações explícitas do usuário:** refere-se à ligação explícita que deve existir entre uma ação do usuário e um processamento do sistema. A interface explicitamente comandada sempre solicita uma ação explícita do usuário de validação global em um formulário para entrada de diversos dados ou parâmetros; separa as ações de seleção de uma opção e de ativação dessa opção quando se referir a um tratamento demorado (exemplo: tela

- de assistente para novas conexões do windows); não coloca o usuário diante de comandos de dupla repercussão (por exemplo, salvar + fechar).
- Controle do usuário: em realização de ações longas seqüenciais e de tratamento demorado, os usuários devem estar no controle dos acontecimentos. Cada possível ação do usuário deve ser antecipada e as opções apropriadas devem ser oferecidas. Em uma interface controlada pelo usuário: a) o cursor não se desloca de um campo a outro em um formulário como efeito colateral das entradas dos usuários (validação [enter]) ou do preenchimento completo de um dado de comprimento controlado (campo senha, por exemplo); ele o faz como efeito do comando explícito de tabulação ([tab]); b) o usuário encontra as opções para comandar o avanço, o recuo, a interrupção, a retomada ou a finalização de um diálogo seqüencial (exemplo: tela de artigo publicado em periódico do currículo lattes); c) o usuário encontra as opções para comandar a interrupção, a retomada ou a finalização de tratamentos demorados.
 - o Adaptabilidade: qualidade esperada em sistemas em que o público-alvo é vasto e variado. Para que todos tenham direito ao mesmo nível de usabilidade, a interface deve propor maneiras variadas de realizar uma tarefa, deixando ao usuário a liberdade de escolher e dominar uma delas no curso de seu aprendizado. Deve permitir que o usuário adapte as apresentações e estilos de diálogo a suas necessidades. Os sub-critérios são:
 - Flexibilidade: envolve duas qualidades diferenciadas – a flexibilidade estrutural e a personalização. Uma interface estruturalmente flexível fornece aos usuários diferentes maneiras de realizar a entrada de dados (por digitação, por seleção); diferentes caminhos para chegar a uma funcionalidade freqüentemente utilizada (ícone na barra de ferramenta, opção em um painel de menu, atalho de teclado) e diferentes opções de formato de arquivos e de unidades para os dados (exemplo: telas de configuração de cores do MS Office, nas quais o usuário pode selecionar uma cor padronizada ou definir uma outra personalizada digitando seu valor no sistema RGB ou clicando sobre um ponto na área de cores). Uma interface personalizável oferece a possibilidade de o usuário personalizar as telas, inserindo ou retirando ícones, dados ou comandos (exemplo: formatos de apresentação dos arquivos/diretórios pelo gerenciador de arquivos Explorer); definir seqüências de ações automáticas (macros); alterar os valores default oferecidos pelo sistema.
 - Consideração da experiência do usuário: uma interface que considere a experiência do usuário fornece aos especialistas atalhos que permitem acesso rápido às funções do sistema e fornece aos usuários intermitentes diálogos passo a passo.
 - o Gestão de erros: esse critério se aplica a todas as situações. A gestão de erros diz respeito a todos os mecanismos que permitem evitar ou reduzir a ocorrência de erros que favoreçam sua correção. Seus sub-critérios são:
 - Proteção contra os erros: diz respeito aos mecanismos empregados para detectar e prevenir os erros de entradas de dados ou de comandos, e impedir que ações de conseqüências desastrosas e/ou não recuperáveis ocorram. Uma interface que protege a interação contra erros informa ao usuário sobre o risco de perda de dados não-gravados ao final de uma sessão de trabalho; não oferece um comando destrutivo como opção *default* e detecta os erros já no momento da digitação de uma entrada individual em vez de fazê-lo apenas no momento da validação do formulário inteiro.

- Qualidade das mensagens de erro: refere-se à pertinência, à legibilidade e à exatidão da informação dada ao usuário sobre a natureza do erro cometido e sobre as ações a serem executadas para corrigi-lo. Uma boa mensagem de erro indica ao usuário a razão ou a natureza do erro cometido, o que ele fez de errado, o que deveria ter feito e o que deve fazer para sair da situação de erro; é orientada para a tarefa, emprega termos específicos e é breve; tem um tom neutro (nem reprovador e nem humorístico).
- Correção dos erros: diz respeito aos meios colocados à disposição do usuário com o objetivo de permitir a correção de seus erros. Há facilidade na correção de erros quando a interface fornece funções *desfazer* e *refazer*; fornece a possibilidade de o usuário refazer apenas a parte errada de uma entrada (indica o dado errado em um formulário, mantendo todos os outros intactos); fornece ligação direta entre o relatório de erro e o local onde ele se produz.
- Homogeneidade/consistência: critério que se aplica de forma geral, mas em particular quando os usuários são novatos ou intermitentes. Este critério refere-se à forma na qual as escolhas no projeto da interface (códigos, denominações, formatos, procedimentos, etc) são conservadas idênticas em contextos idênticos e diferentes para contextos diferentes. Em uma interface homogênea os códigos e denominações são definidos pelos mesmos critérios em contextos idênticos; a distribuição, a apresentação e a denominação dos objetos nas telas são padronizadas; a sintaxe dos procedimentos é padronizada (utiliza os mesmos meios para obter os mesmos resultados).
- Significado de códigos e denominações: critério que se aplica de forma geral, mas em particular quando os usuários são novatos ou intermitentes. Em uma interface significativa os nomes de funções e objetos de interação são familiares para os usuários; os códigos são representativos do conteúdo que veiculam e são distintos (por exemplo: M – masculino / F – feminino, em vez de 1 – homens e 2 – mulheres); as abreviações são de imediata interpretação.
- Compatibilidade: diz respeito ao grau de similaridade entre diferentes sistemas que são executados em um mesmo ambiente operacional (windows, mac, openlook). Trata-se de um tipo de consistência externa entre aplicativos de um mesmo ambiente. Em uma interface compatível a transferência de informações do contexto da tarefa para o do sistema é mais rápida e eficaz; os procedimentos e as tarefas são organizados de maneira a respeitar expectativas ou costumes do usuário; as traduções, as transposições, as interpretações ou referências à documentação são minimizadas (as telas são compatíveis com os documentos em papel, as denominações de comandos são compatíveis com o vocabulário do usuário, etc.); a informação é apresentada de forma diretamente utilizável.

Dentre algumas recomendações ergonômicas para Interface Homem-Computador (IHC), coletadas da ISO 9241 (Requisitos Ergonômicos para o trabalho de escritório informatizado), serão apresentadas algumas referentes a cores utilizadas na interface. Os estereótipos naturais mais populares indicam os seguintes empregos para as cores:

- Vermelho: deve ser utilizada para perigo, alarme, atenção, alerta, calor e comandos de interrupção;
- Amarelo: para advertências, teste e lentidão.
- Verde: para passagem livre, normalidade, vegetação e segurança.
- Laranja: para valor-limite e radiação;

- Azul: para frio, água, céu e calma;
- Cinza: para inatividade, neutralidade.

É importante lembrar que deve ser evitada a codificação de significado por meio de cores que não sejam vistas por pessoas que sofrem de daltonismo, uma alteração genética caracterizada pela falta de um tipo de célula perceptiva para as cores vermelha e/ou verde e/ou azul. O tipo de daltonismo mais freqüente impõe dificuldade para que os portadores façam a distinção entre o verde e o vermelho.

Recomenda-se que sejam usadas poucas cores, sendo cores neutras, com o mesmo brilho e que as cores brilhantes sejam usadas com cautela. Uma forma interessante de usar as cores é explorar as sensações que estas causam sobre as pessoas: o verde descansa, o vermelho atrai a atenção e pode causar irritação, o azul dá sono e o amarelo desperta.

Quanto às recomendações sobre o emprego de fontes, aconselha-se não usar serifa (caracterizada por uma terminação saliente nos caracteres) para vídeos de baixa resolução. Mas, serifas devem ser empregadas em textos longos, como forma de facilitar o reconhecimento rápido dos caracteres. Já nos títulos e rótulos curtos, deve-se empregar fontes sem serifa. Não utilize fontes menores que 12 pontos para telas e menores que 10 pontos para material impresso. Limite o uso de fontes diferentes para textos em até dois tipos. Evite fontes muito grandes que “gritem” com o usuário. Evite textos só com maiúsculas e não exagere com o sublinhado, o negrito e o itálico. As fontes mais populares e indicadas a serem usadas são:

- Arial: para títulos e cabeçalhos de documentos;
- Avant Garde: grandes títulos;
- Courier: documentos impressos, cartas padronizadas, correspondência;
- Helvética: relatórios, títulos de capítulos, de seções, códigos de programas;
- Letter gothic: texto que deve ser simples e claro;
- Romano: correio padronizado;
- Times: documentos diversos, de múltiplo uso, comentários em programas;
- Ultraback: etiquetas de embalagens.

A4.16 – PROPOSTA TÉCNICA

ID documento: **Data:** / / **Versão :**

Responsável pelo documento:

ID Projeto:

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

1. DADOS DA EMPRESA DE DESENVOLVIMENTO

Razão Social:

CNPJ:

Endereço:

Telefones:

Contato:

2. ID DO DOCUMENTO DE REQUISITOS

A4.17 – RELATÓRIO DE ACOMPANHAMENTO DO PROJETO

ID documento: Data: / / Versão :

Responsável pelo documento:

ID Projeto:

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

DADOS DO RELATÓRIO

Data da avaliação	Entrega avaliada	Resultado da avaliação	Responsáveis	Ações Corretivas

A4.18 –RELATÓRIO DE ANÁLISE DO CONTRATO

ID documento:

Data: / /

Responsável pelo documento:

ID Contrato:

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

ALTERAÇÕES SOLICITADAS PELO CLIENTE

ALTERAÇÕES PROMOVIDAS NO CONTRATO

A4.19 – RELATÓRIO DE ANÁLISE DA PROPOSTA

ID documento:

Data: / /

Responsável pelo documento:

ID Proposta:

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

ALTERAÇÕES SOLICITADAS PELO CLIENTE

ALTERAÇÕES PROMOVIDAS NA PROPOSTA

A4.20 – RELATÓRIO DE ANÁLISE DOS REQUISITOS

ID documento:

Data: / /

Responsável pelo documento:

ID Projeto:

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

1. Conflitos relacionados ao Modelo de Negócio

<Inserir aqui os conflitos relacionados ao modelo de negócio (relacionado ao processo de negócio, às regras de negócio) e a solução adotada depois de esclarecido o conflito junto ao cliente.>

2. Conflitos entre requisitos

<Inserir aqui o identificador dos requisitos conflitantes (a partir do Documento de Requisitos), a descrição desses requisitos e a solução adotada depois de esclarecido o conflito junto ao cliente.>

ID. Requisitos conflitantes:

Descrição dos requisitos:

Solução:

ID. Requisitos conflitantes:

Descrição dos requisitos:

Solução:

ID. Requisitos conflitantes:

Descrição dos requisitos:

Solução:

4.21 – RELATÓRIO SOBRE GARANTIA DE QUALIDADE

ID documento:

Data: / /

Responsável pelo documento:

ID Projeto:

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

Avaliações

Data da avaliação	Objetivo avaliado (produto ou processo. No caso de produto, discriminar)	Responsáveis pela avaliação	Não conformidades	Sugestões de melhoria	Data de comunicação aos interessados

A4.22 – RELATÓRIO DE VIABILIDADE DO PROJETO

ID documento: _____ **Data:** / / _____ **Versão :** _____
Responsável pelo documento: _____

ID Projeto: _____

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

<Apresentar a alternativa definida, considerando as viabilidades técnica, econômica e legal, destacando as vantagens/desvantagens da alternativa com base na relação custo X benefício.>

1 VIABILIDADE TÉCNICA E ECONÔMICA/FINANCEIRA

Tabela de Custos

COST DRIVER			
Definição	Valor Unitário	Qtde	Total
		TOTAL	R\$ -

ESTIMATIVAS DE AQUISIÇÃO			
Produto	Data Solicitação	Data Previsão Recebimento	Preço
		TOTAL EM AQUISIÇÃO	R\$ -

CUSTO OPERACIONAL GERAL DO PRODUTO

2 RELAÇÃO CUSTO X BENEFÍCIO

3 SUGESTÃO DE PREÇO

4 CRONOGRAMA DE RECEBIMENTOS

Data de Recebimento	Valor

Apêndice: Fluxo de caixa do projeto

ID:

Fonte: Wikipedia

http://pt.wikipedia.org/wiki/Fluxo_de_caixa

<http://pt.wikipedia.org/wiki/Dre>

A Demonstração do Resultado do Exercício (DRE) é uma demonstração contábil dinâmica que se destina a evidenciar a formação do resultado líquido em um exercício, através do confronto das receitas, custos e despesas, apuradas segundo o princípio contábil do regime de competência.

A DRE oferece uma síntese financeira dos resultados operacionais e não operacionais de uma empresa em certo período. Embora sejam elaboradas anualmente para fins de legais de divulgação, em geral são feitas mensalmente para fins administrativos e trimestralmente para fins fiscais.

Receita é a entrada monetária que ocorre em uma Entidade (Contabilidade) ou patrimônio (Economia), em geral sob a forma de dinheiro ou de créditos representativos de direitos.

Custos são medidas monetárias dos sacrifícios financeiros com os quais uma organização, uma pessoa ou um governo, têm de arcar a fim de atingir seus objetivos, sendo considerados esses ditos objetivos, a utilização de um produto ou serviço qualquer, utilizados na obtenção de outros bens ou serviços. Custos são medidas monetárias resultantes da aplicação de bens e serviços na produção de outros bens e serviços durante o processo de fabricação.

Despesa, para a Contabilidade, é o gasto necessário para a obtenção de receita. As Despesas são gastos que não se identificam com o processo de transformação ou produção dos bens e produtos.

As despesas estão relacionadas aos valores gastos com a estrutura administrativa e comercial da empresa. Ex: aluguel, salários e encargos, pró-labore, telefone, propaganda, impostos, comissões de vendedores etc. Elas ainda são classificadas em fixas e variáveis, sendo as fixas aquelas cujo valor a ser pago não depende do volume, ou do valor das vendas, enquanto que as variáveis são aquelas cujo valor a ser pago está diretamente relacionado ao valor vendido.

Em Finanças, o **fluxo de caixa** (designado em inglês por "cash flow"), refere-se ao montante de caixa recebido e gasto por uma empresa durante um período de tempo definido, algumas vezes ligado a um projeto específico.

Na Contabilidade, uma projeção de fluxo de caixa demonstra todos os pagamentos (direito) e recebimentos esperados em um determinado período de tempo. O controlador de fluxo de caixa necessita de uma visão geral sobre todas as funções da empresa, como: pagamentos, recebimentos, compras de matéria-prima, compras de materiais secundários, salários e outros, por que é necessário prever o que se poderá gastar no futuro dependendo do que se consome hoje.

O fluxo de caixa é uma ótima ferramenta para auxiliar o administrador de determinada empresa nas tomadas de decisões. É através deste "mapa" que os custos fixos e variáveis ficam evidentes, permitindo-se desta forma um controle efetivo sobre determinadas questões empresariais.

A4.23 – ROTEIRO DE PROSPECÇÃO

ID documento:

Data: / /

Versão :

Responsável pelo documento:

HISTÓRICO DE REVISÕES

Data de criação/atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

<Devem ser descritos os passos a serem seguidos para a prospecção do cliente, de acordo com as abordagens definidas no item 4 (plano de ação) do “Plano de Marketing”.>

A4.24 – SISTEMA DE GESTÃO DE CONHECIMENTO

ID documento:

Data: / /

Responsável pelo documento:

HISTÓRICO DE REVISÕES

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

1. Infra-estrutura

< Aqui é descrita a infra-estrutura para apoiar a realização das atividades de gestão de conhecimento. Por exemplo, a utilização de alguma ferramenta >

2. Estratégia

< Aqui é descrita a estratégia para captura, armazenamento, compartilhamento e utilização do conhecimento. >

3. Tipologia do conhecimento

< Aqui é descrita a tipologia do conhecimento, a fim do conhecimento ser classificado para facilitar a sua utilização >

A4.25 – BANCO DE DADOS DE CONFIGURAÇÃO

Este *template* é composto por cinco planilhas: histórico de revisões, itens de configuração, controle de mudanças, planejamento de mudanças e *baselines*.

HISTÓRICO DE REVISÕES DO DOCUMENTO

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

Planilha – Histórico de Revisões

Id. Item Configuração	DocReq_EMP			
Responsável	Engenheiro de Requisitos (Pedro)			
Data Criação/Atualização	01/10/08			
Versões do IC	V0.1			
Status da versão do IC	em desenvolvimento			
Id. solicitação de mudança	Mudança_EMP_0001			
Data de início da realização da mudança	5/10/2008			
Data de término da realização da mudança	6/10/2008			
Descrição da Mudança Ocorrida	Foram adicionados novos requisitos			
Verificação/Validação				
Comentários				

Id. Item Configuração	PlanoProj_EMP			
Responsável				
Data Criação/Atualização				
Versões do IC	V0.1			
Status da versão do IC	em desenvolvimento	baselined	liberada	
Id. solicitação de mudança				
Data de início da realização da mudança				
Data de término da realização da mudança				
Descrição da Mudança Ocorrida				
Verificação/Validação				
Comentários				

Planilha – Itens de Configuração

Id. Solicitação de Mudança	Descrição da Mudança a ser realizada	Tipo de Mudança (simples ou complexa)	Origem (interna ou externa)	Data de solicitação	Solicitante	Autorização (sim/não)	Data da autorização/não autorização	Quem autorizou/ não autorizou	Id. do Contrato

Planilha – Controle de Mudanças

Id. Solicitação de Mudança	Responsável pelo planejamento da mudança	Esforço Total estimado	Data Estimada do Início da realização da mudança	Data Estimada do Término da realização da mudança	Colaboradores estimados	ID Cronograma	ID dos itens de configuração a serem afetados	Custo para efetivação da mudança	Alteração do contrato? ("sim"/"não")	Comentários

Planilha – Planejamento de Mudanças

Identificador da Baseline	Baseline_EMP_0001
Gerente de Projeto que autorizou	Ana
Data de autorização/criação	20/10/2008
Status	liberada
Itens de configuração e versões	DocReq_EMP_V0.1
	PlanoProj_EMP_V0.1

Identificador da Baseline	
Gerente de Projeto que autorizou	
Data de autorização/criação	
Status	bloqueada
Itens de configuração e versões	

Planilha – *Baselines*

A4.26 – CONTROLE DE PROJETOS

Este *template* é composto por nove planilhas: histórico de revisões, identificação de projetos, acompanhamento de projetos, identificação de recursos, alocação de recursos, identificação de colaboradores, alocação de colaboradores, notas fiscais e duplicatas.

HISTÓRICO DE REVISÕES DO DOCUMENTO

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

Planilha – Histórico de Revisões

ID Projeto	Descrição	Pasta Física	Pasta Lógica
projeto 1		sim	sim
projeto 2			
projeto 3			

Planilha – Identificação de Projetos

CONTROLE DE PROJETOS

ID Projeto	Situação	Data Inicialização	Data Formalização	Fase	Ciclo de Desenvolvimento	Data Início da Fase	Data Término da Fase
projeto 1	formalizado			Elaboração	It.01		

CONTROLE DE PROJETOS

ID Projeto	Recurso1	Recurso2	Recurso3	Recurso4	Recurso5	Colaborador1	Hs/Trabalho Colaborador1	Colaborador2	Hs/Trabalho Colaborador2	Colaborador3	Hs/Trabalho Colaborador3	Colaborador4	Hs/Trabalho Colaborador4
projeto 1	recurso2	recurso5	recurso2	recurso1	recurso6	colaborador4		colaborador3					

CONTROLE DE PROJETOS

ID Projeto	Colaborador5	Hs/Trabalho Colaborador5	ID Contrato
projeto 1			

Planilha – Acompanhamento de Projetos

Id. Recurso	Descrição	Tipo
	recurso1	hardware
	recurso2	hardware
	recurso3	plataforma
	recurso4	

Planilha – Identificação de Recursos

Recurso	Data de Alocação	Previsão de término da alocação	Status	Id. Projeto
recurso1			reservado	projeto 1

Planilha – Alocação de Recursos

Id. Colaborador	Descrição	Categoria de Conhecimento	Conhecimento	Nível
	colaborador1	Design		Junior
	colaborador2	Requisitos	elicitar requisitos	Trainee
	colaborador3			
	colaborador4			
	colaborador5			

Planilha – Identificação de Colaboradores

Colaborador	Data de Alocação	Previsão de término da alocação	Status	Id. Projeto
colaborador1			alocado	projeto 2
colaborador3				

Planilha – Alocação de Colaboradores

Cliente	Data de emissão	Valor	Valor da Aliquota	Valor Total da Nota	ID Projeto
nome do cliente/empresa3		R\$ 5.000,00		R\$ 5.000,00	projeto 1

Planilha – Notas Fiscais

Nr da Duplicata	Data de Vencimento	Valor	Nº da NF	Status
1	10/10/2008	R\$ 2.500,00	1	paga
2	10/11/2008	R\$ 2.500,00	1	emitida

Planilha – Duplicatas

A4.27 – CONTROLE DE TAREFAS

Este *template* é composto por duas planilhas: histórico de revisões e lista de tarefas.

HISTÓRICO DE REVISÕES DO DOCUMENTO

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

Planilha – Histórico de Revisões

ID Projeto:

Tarefas	Tempo máximo (de acordo com o estabelecido no cronograma)	Início	Qtde de hs/trabalhadas no dia	Término
Tarefa1	16 hs	22/10/2008	4 hs	
		23/10/2008	8 hs	
		24/10/2008	4 hs	24/10/2008
Tarefa 2				

Planilha – Lista de Tarefas

A4.28 – CRONOGRAMA

Este *template* é composto por duas planilhas: histórico de revisões e cronograma.

HISTÓRICO	DE
REVISÕES	DO
DOCUMENTO	

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

Planilha – Histórico de Revisões

	Precedência	Duração (dias ou hs)	Início	Término	Recursos
Ciclo de Desenvolvimento 1					
Fase 1 - Elaboração					
Documento de Requisitos, Modelo de Negócio e Projeto IHM (entregas)					
E01. Detalhar requisitos (pacote de trabalho)					
* Detalhar/atualizar o "Modelo de Negócio"					
* Detalhar/Atualizar os requisitos funcionais do capítulo 2 do Documento de Requisitos e registrar no capítulo 3 deste documento					
* Detalhar/Atualizar os requisitos de interface externa e outros requisitos não funcionais (tais como requisitos de documentação, de manutenção) do capítulo 2 do Documento de Requisitos e registrar no capítulo 3 deste documento					
* Elaborar o Projeto IHM - Caso o perfil dos usuários não tenha sido definido na fase de concepção, criar Projeto IHM (conforme GCf01), obter perfil dos usuários e registrar no "Projeto IHM" - A partir do perfil dos usuários, definir os princípios gerais do projeto (com base em critérios ergonômicos) - Definir/Atualizar o padrão de telas no arquivo "Projeto IHM" - Definir/Atualizar o mapa de navegação no arquivo "Projeto IHM"					
* Criar/Atualizar a matriz de rastreabilidade entre requisitos do cliente e do produto no Documento de Requisitos					
* Criar/Atualizar a matriz de rastreabilidade entre requisitos funcionais e não funcionais do produto no Documento de Requisitos					
Documento de Requisitos, Relatório de Análise dos Requisitos					
E02. Verificar e Validar os requisitos					
Arquitetura do Software, Documento de Requisitos e Modelo de Design					
E03. Definir solução de projeto					
Lista de Tarefas					
E04. Detalhar o cronograma					
Planilha de Teste, Plano de Projeto, Cronograma					
E05. Definir técnicas e critérios para V&V					
Arquitetura do Software					
E08. Conduzir revisão do milestone: Arquitetura do Software					

Planilha – Cronograma

A4.29 – FLUXO DE CAIXA

Este *template* é composto por dez planilhas: histórico de revisões, investimentos, despesas fixas, mão de obra, despesas variáveis, impostos, receitas, DRE, projeção de fluxo de caixa e demonstrativo de receita e despesas.

HISTÓRICO DE REVISÕES DO DOCUMENTO

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

Planilha – Histórico de Revisões

Descrição	Qtde	Valor Unitário	Valor Total
Infra-estrutura necessária			
Hardware			
Software			
Tempo necessário para desenvolvimento do produto padrão (se for o caso)			
Outros tipos de investimento inicial			
TOTAL		R\$	-

Planilha – Investimentos

Descrição	Mês 1	Mês 2	Mês 3	Mês 4	Mês 5	Mês 6	Mês 7	Mês 8	Mês 9	Mês 10	Mês 11	Mês 12
PRO-LABORE DOS SÓCIOS												
Água, luz, telefone												
Aluguel, condomínio, IPTU												
Escritório de Contabilidade												
Limpeza												
Tarifas Bancárias												
Marketing & Publicidade												
Internet												
Treinamentos e Viagens												
Manutenção e Conservação												
Total	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -

Planilha – Despesas Fixas

Descrição	Valor hr/trab	Hs/trab	Salário	% de Encargos	Encargos	Salário+Encargos	mês 1		mês 2	
							Qtde	Total	Qtde	Total
Programador			R\$ -		R\$ -	R\$ -		R\$ -		R\$ -
Engenheiro de Requisitos			R\$ -		R\$ -	R\$ -		R\$ -		R\$ -
Arquiteto de Software			R\$ -		R\$ -	R\$ -		R\$ -		R\$ -
Especialista em Teste			R\$ -		R\$ -	R\$ -		R\$ -		R\$ -
Gerente de Projetos			R\$ -		R\$ -	R\$ -		R\$ -		R\$ -
TOTAL			R\$ -		R\$ -	R\$ -		R\$ -		R\$ -

Planilha – Mão de obra

Descrição	Mês 1	Mês 2	Mês 3	Mês 4	Mês 5	Mês 6	Mês 7	Mês 8	Mês 9	Mês 10	Mês 11	Mês 12
Horas consultoria terceirizada												
Horas desenvolvimento terceirizado												
Comissão												
Total	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -

Planilha – Despesas Variáveis

Descrição	Alíquota	Incidência
Impostos Federais		receita
Impostos Estaduais		receita
Impostos Municipais		receita

Planilha – Impostos

Descrição	Mês 1	Mês 2	Mês 3	Mês 4	Mês 5	Mês 6
Montante do Valor relativo às horas de customização (produto padrão)						
Montante do Valor relativo às horas de desenvolvimento do Produto padrão						
Montante do Valor relativo às horas de desenvolvimento (on-demand)						

Planilha – Receitas

DEMONSTRAÇÃO DE RESULTADO

	Mês 1	Mês 2	Mês 3	Mês 4	Mês 5
Receita Bruta	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -
Montante do Valor relativo às horas de customização (produto padrão)	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -
Montante do Valor relativo às horas de desenvolvimento do Produto padrão	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -
Montante do Valor relativo às horas de desenvolvimento (on-demand)	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -
(-) Impostos	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -
(-) Comissão	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -
(-) Demais Despesas Variáveis	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -
(=) Receita Líquida	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -
(-) Investimento Inicial	R\$ -				
(-) Custo do trabalho de customização/desenvolvimento	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -
(-) Custo do desenvolvimento do produto padrão					
(=) Margem Contribuição	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -
(-) Despesas Fixas	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -
(=) Resultado Final	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -

Planilha – DRE

ID PROJETO

	jan	fev	mar	abr	mai	jun
Receitas						
Montante do Valor relativo às horas de customização (produto padrão)						
Montante do Valor relativo às horas de desenvolvimento do Produto padrão						
Montante do Valor relativo às horas de desenvolvimento (on-demand)						
Faturamento	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -
Despesas						
Impostos						
Comissão						
Demais Despesas Variáveis						
Investimento Inicial						
Custo do trabalho de customização/desenvolvimento						
Custo do desenvolvimento do produto padrão						
Despesas Fixas						
Gasto Total	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -
Saldo de Caixa	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -	R\$ -

Planilha – Projeção de Fluxo de Caixa

DEMONSTRATIVO DE RECEITAS E DESPESAS DO ANO DE 2008						
	Janeiro		Fevereiro		Março	
	Valor	% Relação (Receita X Despesa)	Valor	% Relação (Receita X Despesa)	Valor	% Relação (Receita X Despesa)
Receita						
Venda de Software	5.000,00		2.300,00		3.000,00	
Mensalidades	2.000,00		2.300,00		2.600,00	
	7.000,00		4.600,00		5.600,00	
Despesas						
PRO-LABORE DOS SÓCIOS	700,00	10,00	900,00	19,57	1.200,00	21,43
Água, luz, telefone	300,00	4,29	300,00	6,52	300,00	5,36
Aluguel, condomínio, IPTU	200,00	2,86	200,00	4,35	200,00	3,57
Escritório de Contabilidade	30,00	0,43	30,00	0,65	30,00	0,54
Limpeza	20,00	0,29	20,00	0,43	20,00	0,36
Tarifas Bancárias	50,00	0,71	50,00	1,09	50,00	0,89
Marketing & Publicidade	30,00	0,43	30,00	0,65	30,00	0,54
Internet	50,00	0,71	50,00	1,09	50,00	0,89
Treinamentos e Viagens	60,00	0,86	60,00	1,30	60,00	1,07
Manutenção e Conservação	230,00	3,29	230,00	5,00	230,00	4,11
Folha de Pagamento	800,00	11,43	800,00	17,39	800,00	14,29
Investimentos	0,00	0,00	200,00	4,35	0,00	0,00
	1.670,00	23,86	1.870,00	40,65	2.170,00	38,75
Lucro	5.330,00		2.730,00		3.430,00	

Planilha – Demonstrativo de Receitas e Despesas

A4.30 – LIÇÕES APRENDIDAS E CONHECIMENTO

Este *template* é composto por duas planilhas: lições aprendidas e conhecimento.

Lições Aprendidas	TIPO	
	Área de conhecimento	Elemento

Planilha – Lições Aprendidas

Conhecimento	TIPO	
	Área de conhecimento	Elemento

Planilha – Conhecimento

Exemplo de Tipologia

Área de conhecimento
Engenharia de Requisitos
Planejamento de Projeto
Monitoramento e Controle de Projeto
Garantia da Qualidade
Medição
Design
Codificação
Testes
Revisões Formais
Estimativas
Gestão de Mudanças e Configuração
Gestão de Requisitos

Elementos
Templates
Atividades
Tarefas
Recursos

A4.31 – LISTA DE CONTATOS

Este *template* é composto por três planilhas: histórico de revisões, lista de contatos e controle de visitas.

HISTÓRICO	DE
REVISÕES	DO
DOCUMENTO	

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

Planilha – Histórico de Revisões

Contatos Validados	Endereço Completo	Telefones	E-mails para contato	Contato chave dentro da empresa	Gargalos e necessidades da empresa	Atividades e Soluções Utilizadas na Empresa e nível de satisfação
nome do cliente/empresa1						
nome do cliente/empresa2						
nome do cliente/empresa3						
nome do cliente/empresa4						

Planilha – Lista de Contatos

Contatos Validados	Data de E-mail marketing	Data do Contato Telefônico	Interesse por proposta	Data da Visita	Data de Solicitação pelo cliente da Proposta	Data de Envio da Proposta para o cliente	Data de Fechamento
nome do cliente/empresa	25/08/08	30/08/08	sim	05/09/08	05/09/08	07/09/08	cancelado - 20/09/2008

Planilha – Controle de Visitas

A4.32 – MEMÓRIA ORGANIZACIONAL

Este *template* é composto pela planilha: memória organizacional.

Conhecimento	TIPO	
	Area de conhecimento	Elemento

Planilha – Memória Organizacional

Exemplo de Tipologia

Área de conhecimento
Engenharia de Requisitos
Planejamento de Projeto
Monitoramento e Controle de Projeto
Garantia da Qualidade
Medição
Design
Codificação
Testes
Revisões Formais
Estimativas
Gestão de Mudanças e Configuração
Gestão de Requisitos

Elementos
Templates
Atividades
Tarefas
Recursos

A4.33 – PLANILHA DE TESTE

Este *template* é composto por quatro planilhas: histórico de revisões, casos de teste de unidade, casos de teste de integração e casos de teste de validação.

HISTÓRICO DE REVISÕES DO DOCUMENTO

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

Planilha – Histórico de Revisões

ID Projeto:

Critério de Teste:		
Casos de Teste Definidos	Resultado Esperado	Resultado Obtido

Critério de Teste:		
Casos de Teste Definidos	Resultado Esperado	Resultado Obtido

Testes de regressão		
Casos de Teste Definidos	Resultado Esperado	Resultado Obtido

Planilha – Casos de Teste de Unidade, de Teste de Integração e de Teste de Validação

A4.34 – PLANO DE AÇÃO

Este *template* é composto por três planilhas: histórico de revisões, plano de ação e acompanhamento.

HISTÓRICO DE REVISÕES DO DOCUMENTO

Data de criação/ atualização	Descrição da(s) Mudança(s) Ocorrida(s)	Autor	Versão do Documento	ID. Solicitação de Mudança

Planilha – Histórico de Revisões

PLANO DE AÇÃO

Plano de Marketing Relacionado:

Planejamento de Atividades	Jan	Fev	Mar	Abr	Mai	Jun	Total
Email marketing	Ligações estimadas com sucesso						0
Contato Telefônico	Ligações estimadas com sucesso						0
Visita	Ligações estimadas com sucesso						0
Proposta	Ligações estimadas com sucesso						0
Fechamento	Ligações estimadas com sucesso						0

Custos - Recursos	Jan	Fev	Mar	Abr	Mai	Jun	Total
Email marketing	Ligações estimadas com sucesso						R\$ -
Contato Telefônico	Ligações estimadas com sucesso						R\$ -
Visita	Ligações estimadas com sucesso						R\$ -
Proposta	Ligações estimadas com sucesso						R\$ -
Fechamento	Ligações estimadas com sucesso						R\$ -

Custo - Pessoas	Jan	Fev	Mar	Abr	Mai	Jun	Total Hs	Total R\$
Nome1							0	R\$ -
Nome2							0	R\$ -
Nome3							0	R\$ -
Nome4							0	R\$ -
Total	0	0	0	0	0	0	0	R\$ -

Pessoas	Valor hr/trab
Nome1	
Nome2	
Nome3	
Nome4	

Total Geral	R\$ -
--------------------	-------

Planilha – Plano de Ação

Atividades	Jan		Fev		Mar		Abr		Mai		Jun		Total		% Real
	Plan	Real	Plan	Real	Plan	Real	Plan	Real	Plan	Real	Plan	Real	Plan	Real	
Email marketing	Ligações com sucesso		0	0	0	0	0	0	0	0	0	0	0	0	0
Contato Telefônico	Ligações com sucesso		0	0	0	0	0	0	0	0	0	0	0	0	0
Visita	Ligações com sucesso		0	0	0	0	0	0	0	0	0	0	0	0	0
Proposta	Ligações com sucesso		0	0	0	0	0	0	0	0	0	0	0	0	0
Fechamento	Ligações com sucesso		0	0	0	0	0	0	0	0	0	0	0	0	0

Custos	Jan		Fev		Mar		Abr		Mai		Jun		Total		% Real
	Plan	Real	Plan	Real	Plan	Real	Plan	Real	Plan	Real	Plan	Real	Plan	Real	
Email marketing	Ligações com sucesso		0	0	0	0	0	0	0	0	0	0	0	0	0
Contato Telefônico	Ligações com sucesso		0	0	0	0	0	0	0	0	0	0	0	0	0
Visita	Ligações com sucesso		0	0	0	0	0	0	0	0	0	0	0	0	0
Proposta	Ligações com sucesso		0	0	0	0	0	0	0	0	0	0	0	0	0
Fechamento	Ligações com sucesso		0	0	0	0	0	0	0	0	0	0	0	0	0

Planilha – Acompanhamento

Apêndice 5 – Modelo ProcSoftVD (Mapeamento com o CMMI e ISO/IEC 15504)

O Quadro A5.1 apresenta uma legenda com os símbolos (e suas respectivas descrições), utilizados no mapeamento das atividades do Modelo ProcSoftVD com o CMMI-DEV e ISO/IEC 15504-5 (Quadros A5.2 a A5.7).

QUADRO A5.1 – LEGENDA DO MAPEAMENTO DAS ATIVIDADES

Fases
P - Prospecção
CP - Concepção
N - Negociação
E - Elaboração
Ct - Construção
T - Transição
Áreas de Conhecimento
ACQ - Aquisição
GqPP - Garantia de Qualidade de Produto e Processo
GCf - Gestão de Mudanças e Configuração
GCo - Gestão de Conhecimento
CMMI – Capability Maturity Model Integration (DEV – Development)
SP - Specific Practices (práticas específicas)
RD - Requirements Development (Desenvolvimento de Requisitos)
REQM - Requirements Management (Gestão de Requisitos)
PP - Project Planning (Planejamento de Projeto)
VER - Verification (Verificação)
VAL - Validation (Validação)
TS - Technical Solution (Solução Técnica)
MA - Measurement and Analysis (Medição e Análise)
SAM - Supplier Agreement Management (Gestão de Acordo com o Fornecedor)
PMC - Project Monitoring and Control (Monitoramento e Controle de Projeto)
PPQA - Process and Product Quality Assurance (Garantia da Qualidade de Produto e Processo)
PI - Product Integration (Integração do Produto)
CM - Configuration Management (Gestão de Configuração)
ISO/IEC 15504-5 - An exemplar Process Assessment Model
BP - Base Practices (Práticas-base)
ENG.1 - Requirements elicitation (Elicitação de Requisitos)
ENG.2 - System requirements analysis (Análise de requisitos de sistema)
ENG.3 - System architectural design (Projeto arquitetural de sistema)
ENG.4 - Software requirements analysis (Análise de requisitos de software)
ENG.5 - Software design (Projeto de Software)
ENG.7 - Software construction (Construção de Software)
ENG.8 - Software testing (Teste de Software)
ENG.9 - System integration (Integração de sistema)
ENG.10 - System testing (Teste de Sistema)

QUADRO A5.1 – LEGENDA DO MAPEAMENTO DAS ATIVIDADES (CONT.)

ENG.11 - Software installation (Instalação do Software)
SUP.2 - Verification (Verificação)
SUP.3 - Validation (Validação)
SUP.7 - Documentation (Documentação)
SUP.8 - Configuration management (Gestão de Configuração)
MAN.3 - Project management (Gestão de Projeto)
MAN.6 - Measurement (Medição)
ACQ.1 - Acquisition preparation (Preparação da Aquisição)
ACQ.2 - Supplier selection (Seleção do Fornecedor)
ACQ.3 - Contract agreement (Acordo)
ACQ.4 - Supplier monitoring (Monitoramento do Fornecedor)
ACQ.5 - Customer acceptance (Aceitação do Cliente)
RIN.3 - Knowledge management (Gestão de Conhecimento)

QUADRO A5.2 – MAPEAMENTO DAS ATIVIDADES DA FASE DE PROSPECÇÃO

P01. Buscar contatos	
CMMI-DEV	ISO/IEC 15504-5
-	-
P02. Prospectar cliente	
CMMI-DEV	ISO/IEC 15504-5
-	-
P03. Visitar cliente	
CMMI-DEV	ISO/IEC 15504-5
RD - SP 1.1 Elicit Needs	ENG.1.BP1: Obtain customer requirements and requests. Obtain and define customer requirements and requests through direct and continuous solicitation of customer and user input.
P04. Criar infra-estrutura do projeto	
CMMI-DEV	ISO/IEC 15504-5
-	-
P05. Controlar custos despendidos na fase de prospecção	
CMMI-DEV	ISO/IEC 15504-5
-	-

QUADRO A5.3 – MAPEAMENTO DAS ATIVIDADES DA FASE DE CONCEPÇÃO

Cp01. Entender o negócio e as necessidades do cliente	
CMMI-DEV	ISO/IEC 15504-5
RD - SP 1.1 Elicit Needs RD - SP 1.2 Develop the Customer Requirements	ENG.1.BP1: Obtain customer requirements and requests. Obtain and define customer requirements and requests through direct and continuous solicitation of customer and user input.
Cp02. Definir escopo do projeto	
CMMI-DEV	ISO/IEC 15504-5
PP - SP 1.1 Estimate the Scope of the Project PP - SP 1.3 Define Project Lifecycle (este objetivo específico é atendido ao ser utilizado esse modelo de processo proposto neste trabalho de pesquisa, como ciclo de vida do projeto) PP - SP 2.7 Establish the Project Plan RD - SP 1.1 Elicit Needs RD - SP 1.2 Develop the Customer Requirements	MAN.3.BP1: Define the scope of work. Identify the project's objectives, motivation and boundaries and define the work to be undertaken by the project. MAN.3.BP2: Define project life cycle. Define a life cycle and strategy for the project, appropriate to its scope, context, magnitude and complexity. (esta prática base é atendida ao ser utilizado esse modelo de processo proposto neste trabalho de pesquisa, como ciclo de vida do projeto) MAN.3.BP10: Establish project plan. Define and maintain project master plan and other relevant plans to cover the project scope and goals, resources, infrastructure, interfaces and communication mechanisms. ENG.1.BP1: Obtain customer requirements and requests. Obtain and define customer requirements and requests through direct and continuous solicitation of customer and user input.
Cp03. Determinar requisitos do produto	
CMMI-DEV	ISO/IEC 15504-5
RD - SP 2.1 Establish Product and Product Component Requirements	ENG.2.BP1: Establish system requirements. Use the stakeholder requirements as the basis for defining the required functions and capabilities of the system and document in a system requirements baseline. ENG.4.BP1: Specify software requirements. Define and prioritize functional and nonfunctional requirements of the software elements of the system and their interfaces and document them in a software requirements specification. Analyze the software requirements for correctness, completeness, consistency, feasibility and testability. Identify any derived requirements.

QUADRO A5.3 – MAPEAMENTO DAS ATIVIDADES DA FASE DE CONCEPÇÃO (CONT.)

Cp04. Verificar e Validar os requisitos	
CMMI-DEV	ISO/IEC 15504-5
RD - SP 3.3 Analyze Requirements RD - SP 3.5 Validate Requirements VER - SP 2.1 Prepare for Peer Reviews VER - SP 2.2 Conduct Peer Reviews VER - SP 2.3 Analyze Peer Review Data VER - SP 3.1 Perform Verification VER - SP 3.2 Analyze Verification Results VAL - SP 2.1 Perform Validation VAL - SP 2.2 Analyze Validation Results	ENG.1.BP2: Understand customer expectations. Ensure that both supplier and customer understand each requirement in the same way. Review with customers their requirements and requests to better understand their needs and expectations and to check the feasibility and appropriateness of their requirements. ENG.1.BP3: Agree on requirements. Obtain agreement across teams on the customer requirements, obtaining the appropriate sign-offs by representatives of all teams and other parties contractually bound to work to these requirements. ENG.2.BP3: Analyze system requirements. Prioritize requirements and analyze the prioritized requirements for correctness, completeness, consistency, feasibility and testability, identifying the necessary elements of the system. Identify changes to the operating environment. SUP.2.BP3: Conduct verification. Verify identified work products according to specified strategy. SUP.2.BP4: Determine actions for verification results. Defects detected by the verification should be identified, recorded and entered into the Problem resolution process (SUP.9). SUP.3.BP3: Perform validation activities. Conduct validation activities using identified techniques, processes, and test cases against requirements and quality standards. SUP.3.BP4: Identify problems. Issues detected by the validation process should be identified, recorded and entered into the Problem resolution management process (SUP.9).
Cp05. Estudar soluções alternativas de arquitetura do software	
CMMI-DEV	ISO/IEC 15504-5
RD - SP 2.2 Allocate Product Component Requirements RD - SP 2.3 Identify Interface Requirements TS - SP 1.1 Develop Alternative Solutions and Selection Criteria TS - SP 1.2 Select Product Component Solutions TS - SP 2.4 Perform Make, Buy, or Reuse Analyses	ENG.3.BP1: Describe system architecture. Establish the top-level system architecture that identifies elements of hardware, software and manual-operations. ENG.3.BP2: Allocate requirements. Allocate all system requirements to the elements of the top-level system architecture. ENG.3.BP3: Define interfaces. Develop and document the internal and external interfaces of each system element. ENG.3.BP4: Verify system architecture. Ensure that the system architecture meets all stakeholder and system requirements. ENG.3.BP5: Evaluate alternative system architectures. Define evaluation criteria for architecture design. Evaluate alternative system architectures according to the defined criteria. Record the rationale for choosing the current system architecture. ENG.4.BP2: Determine operating environment impact. Determine the interfaces between the software requirements and other elements of the operating environment, and the impact that the requirements will have. ENG.5.BP1: Describe software architecture. Transform the software requirements into a software architecture design that describes the top-level structure and identifies its major software elements. ENG.5.BP2: Define interfaces. Specify and document the external and internal interfaces between the software elements.

QUADRO A5.3 – MAPEAMENTO DAS ATIVIDADES DA FASE DE CONCEPÇÃO (CONT.)

Cp06. Definir/Redefinir cronograma do projeto	
CMMI-DEV	ISO/IEC 15504-5
PP - SP 2.1 Establish the Budget and Schedule PP - SP 2.7 Establish the Project Plan	MAN.3.BP5: Define project activities and tasks. Identify project activities and tasks according to defined project lifecycle, and define dependencies between them. MAN.3.BP7: Define project schedule. Allocate resources to activities and determine the sequence and schedule of performance of activities within the project. MAN.3.BP10: Establish project plan. Define and maintain project master plan and other relevant plans to cover the project scope and goals, resources, infrastructure, interfaces and communication mechanisms.
Cp07. Determinar estimativas de esforço e tempo	
CMMI-DEV	ISO/IEC 15504-5
PP - SP 1.2 Establish Estimates of Work Product and Task Attributes PP - SP 1.4 Determine Estimates of Effort and Cost PP - SP 2.7 Establish the Project Plan MA - SP 2.1 Collect Measurement Data MA - SP 2.2 Analyze Measurement Data MA - SP 2.3 Store Data and Results	MAN.3.BP4: Determine and maintain estimates for project attributes. Define and maintain baselines for project attributes. NOTE 1: Project attributes may include 1) business and quality goals for the project, 2) size and complexity of the project and 3) project effort, schedule and budget. NOTE 2: Project quality goals and risks should be considered when estimating project attributes. See Quality management process (MAN.4) and Risk management process (MAN.5) for details. MAN.3.BP10: Establish project plan. Define and maintain project master plan and other relevant plans to cover the project scope and goals, resources, infrastructure, interfaces and communication mechanisms. MAN.6.BP5: Collect and store measurement data. Identify, collect and store measurement data, including context information necessary to verify, understand, or evaluate the data. MAN.6.BP6: Analyze measurement data. Analyze and interpret measurement data, and develop information products. MAN.6.BP7: Use measurement information products for decision-making. Make accurate and current measurement information products accessible for any decision-making processes for which it is relevant.
Cp08. Verificar a viabilidade técnica e financeira do projeto	
CMMI-DEV	ISO/IEC 15504-5
PP - SP 1.4 Determine Estimates of Effort and Cost PP - SP 2.7 Establish the Project Plan MA - SP 2.1 Collect Measurement Data MA - SP 2.2 Analyze Measurement Data MA - SP 2.3 Store Data and Results	MAN.3.BP4: Determine and maintain estimates for project attributes. Define and maintain baselines for project attributes. NOTE 1: Project attributes may include 1) business and quality goals for the project, 2) size and complexity of the project and 3) project effort, schedule and budget. NOTE 2: Project quality goals and risks should be considered when estimating project attributes. See Quality management process (MAN.4) and Risk management process (MAN.5) for details. MAN.3.BP10: Establish project plan. Define and maintain project master plan and other relevant plans to cover the project scope and goals, resources, infrastructure, interfaces and communication mechanisms. MAN.6.BP5: Collect and store measurement data. Identify, collect and store measurement data, including context information necessary to verify, understand, or evaluate the data. MAN.6.BP6: Analyze measurement data. Analyze and interpret measurement data, and develop information products. MAN.6.BP7: Use measurement information products for decision-making. Make accurate and current measurement information products accessible for any decision-making processes for which it is relevant.

QUADRO A5.3 – MAPEAMENTO DAS ATIVIDADES DA FASE DE CONCEPÇÃO (CONT.)

Cp09. Planejar os recursos do projeto	
CMMI-DEV	ISO/IEC 15504-5
PP - SP 2.4 Plan for Project Resources PP - SP 2.6 Plan Stakeholder Involvement PP - SP 2.7 Establish the Project Plan SAM - SP 1.1 Determine Acquisition Type	MAN.3.BP10: Establish project plan. Define and maintain project master plan and other relevant plans to cover the project scope and goals, resources, infrastructure, interfaces and communication mechanisms. ACQ.1.BP1: Establish the need. Establish a need to acquire, develop, or enhance a system, software product or service. ACQ.1.BP2: Define the requirements. Identify the customer/stakeholder requirements for a system and/or software product or service. ACQ.1.BP3: Review requirements. Analyze and validate the defined requirements against the identified needs. Validate the requirements to reduce risk of misunderstanding by the potential suppliers
Cp10. Identificar conhecimentos e habilidades necessárias	
CMMI-DEV	ISO/IEC 15504-5
PP - SP 2.5 Plan for Needed Knowledge and Skills PP - SP 2.6 Plan Stakeholder Involvement PP - SP 2.7 Establish the Project Plan	MAN.3.BP6: Define needs for experience, knowledge and skills. Identify the experience, knowledge and skill requirements of the project and apply them to the selection of individuals and teams.
Cp11. Identificar riscos inerentes ao projeto	
CMMI-DEV	ISO/IEC 15504-5
RD - SP 3.4 Analyze Requirements to Achieve Balance PP - SP 2.2 Identify Project Risks PP - SP 2.7 Establish the Project Plan MA - SP 2.1 Collect Measurement Data MA - SP 2.2 Analyze Measurement Data MA - SP 2.3 Store Data and Results	ENG.2.BP4: Evaluate and update system requirements. Evaluate the impact of proposed changes and new requirements for cost, schedule, risk and technical impact, approve or reject changes and new requirements, and update the system requirements baseline. ENG.4.BP5: Evaluate and update software requirements. Evaluate the requirements with the customer, evaluate the impact of proposed changes for cost, schedule and technical impact, approve or reject changes, and update the software requirements specification. MAN.3.BP10: Establish project plan. Define and maintain project master plan and other relevant plans to cover the project scope and goals, resources, infrastructure, interfaces and communication mechanisms. MAN.6.BP5: Collect and store measurement data. Identify, collect and store measurement data, including context information necessary to verify, understand, or evaluate the data. MAN.6.BP6: Analyze measurement data. Analyze and interpret measurement data, and develop information products. MAN.6.BP7: Use measurement information products for decision-making. Make accurate and current measurement information products accessible for any decision-making processes for which it is relevant.
Cp12. Acompanhar o andamento do projeto	
CMMI-DEV	ISO/IEC 15504-5
PMC - SP 1.1 Monitor Project Planning Parameters PMC - SP 1.2 Monitor Commitments PMC - SP 1.3 Monitor Project Risks	MAN.3.BP12: Monitor project attributes. Monitor project scope, budget, cost, resources and other necessary attributes and document significant deviations of them against the project baseline.
Cp13. Controlar custos despendidos na fase de concepção	
CMMI-DEV	ISO/IEC 15504-5
PMC - SP 1.1 Monitor Project Planning Parameters	MAN.3.BP12: Monitor project attributes. Monitor project scope, budget, cost, resources and other necessary attributes and document significant deviations of them against the project baseline.

QUADRO A5.3 – MAPEAMENTO DAS ATIVIDADES DA FASE DE CONCEPÇÃO (CONT.)

Cp14. Conduzir revisão do milestone: Estudo de Viabilidade do Projeto	
CMMI-DEV	ISO/IEC 15504-5
PMC - SP 1.7 Conduct Milestone Reviews PP - SP 3.1 Review Plans That Affect the Project PPQA - SP 1.1 Objectively Evaluate Processes PPQA - SP 1.2 Objectively Evaluate Work Products and Services PPQA - SP 2.1 Communicate and Ensure Resolution of Noncompliance Issues PPQA - SP 2.2 Establish Records MA - SP 2.1 Collect Measurement Data MA - SP 2.2 Analyze Measurement Data MA - SP 2.3 Store Data and Results MA - SP 2.4 Communicate Results	SUP.1.BP2: Define quality records. Quality records are defined that demonstrate conformance of process and work products to their quality requirements. SUP.1.BP3: Assure the quality of project process activities and project work products. Carry out a series of activities to provide assurance, with the required level of confidence, that the project processes have followed specified standards and that the work products meet the quality requirements. SUP.1.BP4: Identify and record problems and non-conformances. Problems and nonconformances are identified and recorded and then reported to appropriate stakeholders for information and action. SUP.1.BP5: Act on non-conformances. Deviations or non-conformance with agreed requirements or organizational quality goals are analyzed and resolved. MAN.6.BP5: Collect and store measurement data. Identify, collect and store measurement data, including context information necessary to verify, understand, or evaluate the data. MAN.6.BP6: Analyze measurement data. Analyze and interpret measurement data, and develop information products. MAN.6.BP7: Use measurement information products for decision-making. Make accurate and current measurement information products accessible for any decision-making processes for which it is relevant. MAN.6.BP8: Communicate measurement results. Disseminate measurement information products to all parties who will be using them and collect feedback to evaluate the appropriateness for intended use. MAN.6.BP9: Evaluate and communicate information products and measurement activities to process owners. Evaluate information products and measurement activities against the identified information needs and measurement strategy, identify potential improvements in measurements, and communicate any identified potential improvement to the process owners.

QUADRO A5.4 – MAPEAMENTO DAS ATIVIDADES DA FASE DE NEGOCIAÇÃO

N01. Elaborar proposta	
CMMI-DEV	ISO/IEC 15504-5
PP - SP 3.2 Reconcile Work and Resource Levels	ENG.4.BP5: Evaluate and update software requirements. Evaluate the requirements with the customer, evaluate the impact of proposed changes for cost, schedule and technical impact, approve or reject changes, and update the software requirements specification. MAN.3.BP3 Evaluate feasibility of the project. Evaluate the feasibility of achieving the goals of the project with available resources and constraints.
N02. Analisar proposta	
CMMI-DEV	ISO/IEC 15504-5
-	-
N03. Elaborar contrato	
CMMI-DEV	ISO/IEC 15504-5
-	-

QUADRO A5.4 – MAPEAMENTO DAS ATIVIDADES DA FASE DE NEGOCIAÇÃO (CONT.)

N04. Analisar contrato	
CMMI-DEV	ISO/IEC 15504-5
-	-
N05. Definir time do projeto	
CMMI-DEV	ISO/IEC 15504-5
PP - SP 2.5 Plan for Needed Knowledge and Skills PP - SP 2.1 Establish the Budget and Schedule	MAN.3.BP7: Define project schedule. Allocate resources to activities and determine the sequence and schedule of performance of activities within the project. MAN.3.BP6: Define needs for experience, knowledge and skills. Identify the experience, knowledge and skill requirements of the project and apply them to the selection of individuals and teams.
N06. Formalizar início do projeto	
CMMI-DEV	ISO/IEC 15504-5
PP - SP 3.3 Obtain Plan Commitment	MAN.3.BP8: Identify and monitor project interfaces. Identify and agree interfaces of the project with other projects, organizational units and other affected parties and monitor agreed commitments. MAN.3.BP9: Allocate responsibilities. Identify the specific individuals and groups contributing to, and impacted by, the project, allocate them their specific responsibilities, and ensure that the commitments are understood and accepted, funded and achievable.
N07. Acompanhar o andamento do projeto	
CMMI-DEV	ISO/IEC 15504-5
PMC - SP 1.1 Monitor Project Planning Parameters PMC - SP 1.2 Monitor Commitments PMC - SP 1.3 Monitor Project Risks	MAN.3.BP12: Monitor project attributes. Monitor project scope, budget, cost, resources and other necessary attributes and document significant deviations of them against the project baseline.
N08. Controlar custos despendidos na fase de negociação	
CMMI-DEV	ISO/IEC 15504-5
PMC - SP 1.1 Monitor Project Planning Parameters	MAN.3.BP12: Monitor project attributes. Monitor project scope, budget, cost, resources and other necessary attributes and document significant deviations of them against the project baseline.

QUADRO A5.4 – MAPEAMENTO DAS ATIVIDADES DA FASE DE NEGOCIAÇÃO (CONT.)

N09. Conduzir revisão do milestone: Contrato	
CMMI-DEV	ISO/IEC 15504-5
PMC - SP 1.7 Conduct Milestone Reviews PP - SP 3.1 Review Plans That Affect the Project PPQA - SP 1.1 Objectively Evaluate Processes PPQA - SP 1.2 Objectively Evaluate Work Products and Services PPQA - SP 2.1 Communicate and Ensure Resolution of Noncompliance Issues PPQA - SP 2.2 Establish Records MA - SP 2.1 Collect Measurement Data MA - SP 2.2 Analyze Measurement Data MA - SP 2.3 Store Data and Results MA - SP 2.4 Communicate Results	SUP.1.BP2: Define quality records. Quality records are defined that demonstrate conformance of process and work products to their quality requirements. SUP.1.BP3: Assure the quality of project process activities and project work products. Carry out a series of activities to provide assurance, with the required level of confidence, that the project processes have followed specified standards and that the work products meet the quality requirements. SUP.1.BP4: Identify and record problems and non-conformances. Problems and nonconformances are identified and recorded and then reported to appropriate stakeholders for information and action. SUP.1.BP5: Act on non-conformances. Deviations or non-conformance with agreed requirements or organizational quality goals are analyzed and resolved. MAN.6.BP5: Collect and store measurement data. Identify, collect and store measurement data, including context information necessary to verify, understand, or evaluate the data. MAN.6.BP6: Analyze measurement data. Analyze and interpret measurement data, and develop information products. MAN.6.BP7: Use measurement information products for decision-making. Make accurate and current measurement information products accessible for any decision-making processes for which it is relevant. MAN.6.BP8: Communicate measurement results. Disseminate measurement information products to all parties who will be using them and collect feedback to evaluate the appropriateness for intended use. MAN.6.BP9: Evaluate and communicate information products and measurement activities to process owners. Evaluate information products and measurement activities against the identified information needs and measurement strategy, identify potential improvements in measurements, and communicate any identified potential improvement to the process owners.

QUADRO A5.5 – MAPEAMENTO DAS ATIVIDADES DA FASE DE ELABORAÇÃO

E01. Detalhar requisitos	
CMMI-DEV	ISO/IEC 15504-5
RD - SP 2.1 Establish Product and Product Component Requirements RD - SP 3.1 Establish Operational Concepts and Scenarios RD - SP 3.2 Establish a Definition of Required Functionality REQM - SP 1.4 Maintain Bidirectional Traceability of Requirements	ENG.4.BP1: Specify software requirements. Define and prioritize functional and nonfunctional requirements of the software elements of the system and their interfaces and document them in a software requirements specification. Analyze the software requirements for correctness, completeness, consistency, feasibility and testability. Identify any derived requirements.
E02. Verificar e Validar os requisitos	
CMMI-DEV	ISO/IEC 15504-5
RD - SP 3.3 Analyze Requirements RD - SP 3.5 Validate Requirements VER - SP 2.1 Prepare for Peer Reviews VER - SP 2.2 Conduct Peer Reviews VER - SP 2.3 Analyze Peer Review Data VER - SP 3.1 Perform Verification VER - SP 3.2 Analyze Verification Results VAL - SP 2.1 Perform Validation VAL - SP 2.2 Analyze Validation Results	ENG.1.BP2: Understand customer expectations. Ensure that both supplier and customer understand each requirement in the same way. Review with customers their requirements and requests to better understand their needs and expectations and to check the feasibility and appropriateness of their requirements. ENG.1.BP3: Agree on requirements. Obtain agreement across teams on the customer requirements, obtaining the appropriate sign-offs by representatives of all teams and other parties contractually bound to work to these requirements. ENG.2.BP3: Analyze system requirements. Prioritize requirements and analyze the prioritized requirements for correctness, completeness, consistency, feasibility and testability, identifying the necessary elements of the system. Identify changes to the operating environment. SUP.2.BP3: Conduct verification. Verify identified work products according to specified strategy. SUP.2.BP4: Determine actions for verification results. Defects detected by the verification should be identified, recorded and entered into the Problem resolution process (SUP.9). SUP.3.BP3: Perform validation activities. Conduct validation activities using identified techniques, processes, and test cases against requirements and quality standards. SUP.3.BP4: Identify problems. Issues detected by the validation process should be identified, recorded and entered into the Problem resolution management process (SUP.9).

QUADRO A5.5 – MAPEAMENTO DAS ATIVIDADES DA FASE DE ELABORAÇÃO (CONT.)

E03. Definir solução de projeto	
CMMI-DEV	ISO/IEC 15504-5
RD - SP 2.2 Allocate Product Component Requirements RD - SP 2.3 Identify Interface Requirements REQM - SP 1.4 Maintain Bidirectional Traceability of Requirements TS - SP 2.1 Design the Product or Product Component TS - SP 2.2 Establish a Technical Data Package TS - SP 2.3 Design Interfaces Using Criteria	ENG.3.BP2: Allocate requirements. Allocate all system requirements to the elements of the top-level system architecture. ENG.3.BP3: Define interfaces. Develop and document the internal and external interfaces of each system element. ENG.3.BP1: Describe system architecture. Establish the top-level system architecture that identifies elements of hardware, software and manual-operations. ENG.5.BP1: Describe software architecture. Transform the software requirements into a software architecture design that describes the top-level structure and identifies its major software elements. ENG.4.BP2: Determine operating environment impact. Determine the interfaces between the software requirements and other elements of the operating environment, and the impact that the requirements will have. ENG.5.BP2: Define interfaces. Specify and document the external and internal interfaces between the software elements. ENG.5.BP3: Develop detailed design. Decompose the software architectural design into a detailed design for each software element describing all software units to be produced and tested. Document software units and interfaces in a software design document.
E04. Detalhar o cronograma	
CMMI-DEV	ISO/IEC 15504-5
PP - SP 2.1 Establish the Budget and Schedule	MAN.3.BP5: Define project activities and tasks. Identify project activities and tasks according to defined project lifecycle, and define dependencies between them. MAN.3.BP7: Define project schedule. Allocate resources to activities and determine the sequence and schedule of performance of activities within the project.

QUADRO A5.5 – MAPEAMENTO DAS ATIVIDADES DA FASE DE ELABORAÇÃO (CONT.)

E05. Definir técnicas e critérios para V&V	
CMMI-DEV	ISO/IEC 15504-5
PP - SP 2.7 Establish the Project Plan PI - SP 1.1 Determine Integration Sequence PI - SP 1.2 Establish the Product Integration Environment PI - SP 1.3 Establish Product Integration Procedures and Criteria VER - SP 1.1 Select Work Products for Verification VER - SP 1.2 Establish the Verification Environment VER - SP 1.3 Establish Verification Procedures and Criteria VAL - SP 1.1 Select Products for Validation VAL - SP 1.2 Establish the Validation Environment VAL - SP 1.3 Establish Validation Procedures and Criteria	<p>MAN.3.BP10: Establish project plan. Define and maintain project master plan and other relevant plans to cover the project scope and goals, resources, infrastructure, interfaces and communication mechanisms.</p> <p>ENG.4.BP3: Develop criteria for software testing. Use the software requirements to define acceptance criteria for the software product tests. Software product tests should demonstrate compliance with the software requirements.</p> <p>ENG.6.BP1: Develop unit verification procedures. Develop and document procedures and criteria for verifying that each software unit satisfies its design requirements. The verification procedure includes unit test cases, unit test data and code review.</p> <p>ENG.7.BP1: Develop software integration strategy. Develop the strategy for integrating software units considering the software requirements. Identify software items based on the software architecture and define a sequence or order for integrating and testing them.</p> <p>ENG.7.BP2: Develop tests for integrated software items. Describe the tests to be run against each integrated software item, including the verification of the interfaces, indicating software requirements being checked, input data and verification criteria.</p> <p>ENG.7.BP6: Regression test integrated software items. Develop a software regression test strategy for re-testing the integrated software items. If changes are made to software units, designs or requirements, carry out regression testing according to this strategy.</p> <p>ENG.8.BP1: Develop tests for integrated software product. Describe the tests to be run against the integrated software product, indicating software requirements being checked, input data, and verification criteria. The set of tests should demonstrate compliance with the software requirements.</p> <p>ENG.8.BP3: Regression test integrated software. Develop a software regression test strategy for re-testing the integrated software product. If changes are made to software items, carry out regression testing according to the strategy.</p> <p>ENG.9.BP2: Develop tests for system elements. Describe the tests to run against each system element, indicating requirements being checked, input data, system elements needed to perform the test, and verification criteria.</p> <p>ENG.10.BP1: Develop tests for system. Describe the tests to be run against the complete system, indicating system requirements being checked, input data, and validation criteria.</p> <p>SUP.2.BP1: Develop verification strategy. Develop and implement a verification strategy, including verification activities with associated methods, techniques, and tool.</p> <p>SUP.2.BP2: Develop criteria for verification. Develop the criteria for verification of all required work products.</p> <p>SUP.3.BP2: Develop validation criteria. Develop the criteria for validation of all required work products.</p>

QUADRO A5.5 – MAPEAMENTO DAS ATIVIDADES DA FASE DE ELABORAÇÃO (CONT.)

E06. Monitorar e controlar o projeto	
CMMI-DEV	ISO/IEC 15504-5
PMC - SP 1.1 Monitor Project Planning Parameters PMC - SP 1.2 Monitor Commitments PMC - SP 1.3 Monitor Project Risks PMC - SP 1.4 Monitor Data Management PMC - SP 1.5 Monitor Stakeholder Involvement PMC - SP 1.6 Conduct Progress Reviews PMC - SP 2.1 Analyze Issues PMC - SP 2.2 Take Corrective Action PMC - SP 2.3 Manage Corrective Action	MAN.3.BP12: Monitor project attributes. Monitor project scope, budget, cost, resources and other necessary attributes and document significant deviations of them against the project baseline. MAN.3.BP13: Review progress of the project. Regularly report and review the status of the project performance against the project plan. MAN.3.BP14: Act to correct deviations. Take action when project goals are not achieved, to correct deviations from the plan and to prevent recurrence of problems identified in the project. Update project plans accordingly.
E07. Monitorar e Controlar questões financeiras	
CMMI-DEV	ISO/IEC 15504-5
PMC - SP 1.1 Monitor Project Planning Parameters PMC - SP 1.2 Monitor Commitments PMC - SP 1.3 Monitor Project Risks	MAN.3.BP12: Monitor project attributes. Monitor project scope, budget, cost, resources and other necessary attributes and document significant deviations of them against the project baseline.
E08. Conduzir revisão do milestone: Arquitetura do Software	
CMMI-DEV	ISO/IEC 15504-5
PMC - SP 1.7 Conduct Milestone Reviews PP - SP 3.1 Review Plans That Affect the Project PPQA - SP 1.1 Objectively Evaluate Processes PPQA - SP 1.2 Objectively Evaluate Work Products and Services PPQA - SP 2.1 Communicate and Ensure Resolution of Noncompliance Issues PPQA - SP 2.2 Establish Records MA - SP 2.1 Collect Measurement Data MA - SP 2.2 Analyze Measurement Data MA - SP 2.3 Store Data and Results MA - SP 2.4 Communicate Results	SUP.1.BP2: Define quality records. Quality records are defined that demonstrate conformance of process and work products to their quality requirements. SUP.1.BP3: Assure the quality of project process activities and project work products. Carry out a series of activities to provide assurance, with the required level of confidence, that the project processes have followed specified standards and that the work products meet the quality requirements. SUP.1.BP4: Identify and record problems and non-conformances. Problems and nonconformances are identified and recorded and then reported to appropriate stakeholders for information and action. SUP.1.BP5: Act on non-conformances. Deviations or non-conformance with agreed requirements or organizational quality goals are analyzed and resolved. MAN.6.BP5: Collect and store measurement data. Identify, collect and store measurement data, including context information necessary to verify, understand, or evaluate the data. MAN.6.BP6: Analyze measurement data. Analyze and interpret measurement data, and develop information products. MAN.6.BP7: Use measurement information products for decision-making. Make accurate and current measurement information products accessible for any decision-making processes for which it is relevant. MAN.6.BP8: Communicate measurement results. Disseminate measurement information products to all parties who will be using them and collect feedback to evaluate the appropriateness for intended use. MAN.6.BP9: Evaluate and communicate information products and measurement activities to process owners. Evaluate information products and measurement activities against the identified information needs and measurement strategy, identify potential improvements in measurements, and communicate any identified potential improvement to the process owners.

QUADRO A5.6 – MAPEAMENTO DAS ATIVIDADES DA FASE DE CONSTRUÇÃO

Ct01. Codificar componentes	
CMMI-DEV	ISO/IEC 15504-5
TS - SP 3.1 Implement the Design	ENG.6.BP2: Develop software units. Develop and document the executable representations of each software unit. Update test requirements and user documentation.
Ct02. Realizar teste de unidade	
CMMI-DEV	ISO/IEC 15504-5
VER - SP 3.1 Perform Verification MA - SP 2.1 Collect Measurement Data MA - SP 2.3 Store Data and Results	<p>ENG.6.BP4: Verify software units. Verify that each software unit satisfies its design requirements by executing the specified unit verification procedures and document the results.</p> <p>ENG.7.BP4: Test integrated software items. Test each integrated software item on an operational platform or suitable equivalent platform, against the verification criteria, and record the results. Update user documentation as necessary.</p> <p>ENG.7.BP6: Regression test integrated software items. Develop a software regression test strategy for re-testing the integrated software items. If changes are made to software units, designs or requirements, carry out regression testing according to this strategy.</p> <p>ENG.8.BP2: Test integrated software product. Test the integrated software product against the verification criteria, and record the results. Update user documentation as necessary.</p> <p>ENG.8.BP3: Regression test integrated software. Develop a software regression test strategy for re-testing the integrated software product. If changes are made to software items, carry out regression testing according to the strategy.</p> <p>ENG.9.BP4: Test system elements. Test each system element and ensure that it satisfies its requirements, and document the results.</p> <p>ENG.9.BP5: Regression test system elements. If changes are made to system elements, carry out regression testing as defined in the regression test strategy.</p> <p>SUP.2.BP3: Conduct verification. Verify identified work products according to specified strategy.</p> <p>MAN.6.BP5: Collect and store measurement data. Identify, collect and store measurement data, including context information necessary to verify, understand, or evaluate the data.</p>
Ct03. Analisar os resultados de teste	
CMMI-DEV	ISO/IEC 15504-5
VER - SP 3.2 Analyze Verification Results MA - SP 2.2 Analyze Measurement Data	<p>SUP.2.BP4: Determine actions for verification results. Defects detected by the verification should be identified, recorded and entered into the Problem resolution process.</p> <p>MAN.6.BP6: Analyze measurement data. Analyze and interpret measurement data, and develop information products.</p> <p>MAN.6.BP7: Use measurement information products for decision-making. Make accurate and current measurement information products accessible for any decision-making processes for which it is relevant.</p>

QUADRO A5.6 – MAPEAMENTO DAS ATIVIDADES DA FASE DE CONSTRUÇÃO (CONT.)

Ct04. Integrar e testar integração entre componentes	
CMMI-DEV	ISO/IEC 15504-5
PI - SP 1.1 Determine Integration Sequence PI - SP 2.1 Review Interface Descriptions for Completeness PI - SP 2.2 Manage Interfaces PI - SP 3.1 Confirm Readiness of Product Components for Integration PI - SP 3.2 Assemble Product Components PI - SP 3.3 Evaluate Assembled Product Components VER - SP 3.1 Perform Verification MA - SP 2.1 Collect Measurement Data MA - SP 2.3 Store Data and Results	ENG.6.BP4: Verify software units. Verify that each software unit satisfies its design requirements by executing the specified unit verification procedures and document the results. ENG.7.BP1: Develop software integration strategy. Develop the strategy for integrating software units considering the software requirements. Identify software items based on the software architecture and define a sequence or order for integrating and testing them. ENG.7.BP3: Integrate software item. Integrate the software units according to the integration strategy to form a software item. ENG.7.BP4: Test integrated software items. Test each integrated software item on a operational platform or suitable equivalent platform, against the verification criteria, and record the results. Update user documentation as necessary. ENG.7.BP6: Regression test integrated software items. Develop a software regression test strategy for re-testing the integrated software items. If changes are made to software units, designs or requirements, carry out regression testing according to this strategy. ENG.8.BP2: Test integrated software product. Test the integrated software product against the verification criteria, and record the results. Update user documentation as necessary. ENG.8.BP3: Regression test integrated software. Develop a software regression test strategy for re-testing the integrated software product. If changes are made to software items, carry out regression testing according to the strategy. ENG.9.BP3: Integrate system elements. Integrate system elements according to the system integration strategy. ENG.9.BP4: Test system elements. Test each system element and ensure that it satisfies its requirements, and document the results. ENG.9.BP5: Regression test system elements. If changes are made to system elements, carry out regression testing as defined in the regression test strategy. ENG.9.BP7: Build complete system of system elements. Identify and integrate system elements to produce a complete system ready for system testing according to the system integration strategy. ENG.10.BP2: Test integrated system. Test the integrated system and ensure that it satisfies the system requirements, and record the results. ENG.10.BP3: Regression test integrated system. Develop a system regression test strategy for re-testing the system. If changes are made to system elements, carry out regression testing as defined in the system regression test strategy. SUP.2.BP3: Conduct verification. Verify identified work products according to specified strategy. MAN.6.BP5: Collect and store measurement data. Identify, collect and store measurement data, including context information necessary to verify, understand, or evaluate the data.

QUADRO A5.6 – MAPEAMENTO DAS ATIVIDADES DA FASE DE CONSTRUÇÃO (CONT.)

Ct05. Monitorar e controlar o projeto	
CMMI-DEV	ISO/IEC 15504-5
PMC - SP 1.1 Monitor Project Planning Parameters PMC - SP 1.2 Monitor Commitments PMC - SP 1.3 Monitor Project Risks PMC - SP 1.4 Monitor Data Management PMC - SP 1.5 Monitor Stakeholder Involvement PMC - SP 1.6 Conduct Progress Reviews PMC - SP 2.1 Analyze Issues PMC - SP 2.2 Take Corrective Action PMC - SP 2.3 Manage Corrective Action	MAN.3.BP12: Monitor project attributes. Monitor project scope, budget, cost, resources and other necessary attributes and document significant deviations of them against the project baseline. MAN.3.BP13: Review progress of the project. Regularly report and review the status of the project performance against the project plan. MAN.3.BP14: Act to correct deviations. Take action when project goals are not achieved, to correct deviations from the plan and to prevent recurrence of problems identified in the project. Update project plans accordingly.
Ct06. Monitorar e controlar questões financeiras	
CMMI-DEV	ISO/IEC 15504-5
PMC - SP 1.1 Monitor Project Planning Parameters PMC - SP 1.2 Monitor Commitments PMC - SP 1.3 Monitor Project Risks	MAN.3.BP12: Monitor project attributes. Monitor project scope, budget, cost, resources and other necessary attributes and document significant deviations of them against the project baseline.
Ct07. Conduzir revisão do milestone: Capacidade Operacional	
CMMI-DEV	ISO/IEC 15504-5
PMC - SP 1.7 Conduct Milestone Reviews PP - SP 3.1 Review Plans That Affect the Project PPQA - SP 1.1 Objectively Evaluate Processes PPQA - SP 1.2 Objectively Evaluate Work Products and Services PPQA - SP 2.1 Communicate and Ensure Resolution of Noncompliance Issues PPQA - SP 2.2 Establish Records MA - SP 2.1 Collect Measurement Data MA - SP 2.2 Analyze Measurement Data MA - SP 2.3 Store Data and Results MA - SP 2.4 Communicate Results	SUP.1.BP2: Define quality records. Quality records are defined that demonstrate conformance of process and work products to their quality requirements. SUP.1.BP3: Assure the quality of project process activities and project work products. Carry out a series of activities to provide assurance, with the required level of confidence, that the project processes have followed specified standards and that the work products meet the quality requirements. SUP.1.BP4: Identify and record problems and non-conformances. Problems and nonconformances are identified and recorded and then reported to appropriate stakeholders for information and action. SUP.1.BP5: Act on non-conformances. Deviations or non-conformance with agreed requirements or organizational quality goals are analyzed and resolved. MAN.6.BP5: Collect and store measurement data. Identify, collect and store measurement data, including context information necessary to verify, understand, or evaluate the data. MAN.6.BP6: Analyze measurement data. Analyze and interpret measurement data, and develop information products. MAN.6.BP7: Use measurement information products for decision-making. Make accurate and current measurement information products accessible for any decision-making processes for which it is relevant. MAN.6.BP8: Communicate measurement results. Disseminate measurement information products to all parties who will be using them and collect feedback to evaluate the appropriateness for intended use. MAN.6.BP9: Evaluate and communicate information products and measurement activities to process owners. Evaluate information products and measurement activities against the identified information needs and measurement strategy, identify potential improvements in measurements, and communicate any identified potential improvement to the process owners.

QUADRO A5.7 – MAPEAMENTO DAS ATIVIDADES DA FASE DE TRANSIÇÃO

T01. Validar a versão do produto	
CMMI-DEV	ISO/IEC 15504-5
VAL - SP 2.1 Perform Validation MA - SP 2.1 Collect Measurement Data MA - SP 2.3 Store Data and Results	ENG.8.BP2: Test integrated software product. Test the integrated software product against the verification criteria, and record the results. Update user documentation as necessary ENG.10.BP2: Test integrated system. Test the integrated system and ensure that it satisfies the system requirements, and record the results. SUP.3.BP3: Perform validation activities. Conduct validation activities using identified techniques, processes, and test cases against requirements and quality standards. The results of validation activities are recorded. MAN.6.BP5: Collect and store measurement data. Identify, collect and store measurement data, including context information necessary to verify, understand, or evaluate the data.
T02. Analisar e Ajustar a versão do produto	
CMMI-DEV	ISO/IEC 15504-5
VAL - SP 2.2 Analyze Validation Results MA - SP 2.2 Analyze Measurement Data TS - SP 3.1 Implement the Design	ENG.11.BP3: Specify the requirements for adaptation. Specify the requirements for adaptation of the system for its intended environment. MAN.6.BP6: Analyze measurement data. Analyze and interpret measurement data, and develop information products. MAN.6.BP7: Use measurement information products for decision-making. Make accurate and current measurement information products accessible for any decision-making processes for which it is relevant. ENG.11.BP4: Adapt the system. Adapt the system to meet the requirements for operation. SUP.3.BP4: Identify problems. Issues detected by the validation process should be identified, recorded and entered into the Problem resolution management process (SUP.9).
T03. Realizar testes de regressão	
CMMI-DEV	ISO/IEC 15504-5
VER - SP 3.1 Perform Verification VAL - SP 2.1 Perform Validation MA - SP 2.1 Collect Measurement Data MA - SP 2.3 Store Data and Results	SUP.2.BP3: Conduct verification. Verify identified work products according to specified strategy. SUP.3.BP3: Perform validation activities. Conduct validation activities using identified techniques, processes, and test cases against requirements and quality standards. The results of validation activities are recorded. ENG.10.BP3: Regression test integrated system. Develop a system regression test strategy for re-testing the system. If changes are made to system elements, carry out regression testing as defined in the system regression test strategy. MAN.6.BP5: Collect and store measurement data. Identify, collect and store measurement data, including context information necessary to verify, understand, or evaluate the data.

QUADRO A5.7 – MAPEAMENTO DAS ATIVIDADES DA FASE DE TRANSIÇÃO (CONT.)

T04. Entregar/Instalar a versão do produto	
CMMI-DEV	ISO/IEC 15504-5
TS - SP 3.2 Develop Product Support Documentation PI - SP 3.4 Package and Deliver the Product or Product Component	SUP.7.BP5: Develop documents. Develop documents at required process points according to established standards and policy. ENG.11.BP5: Install software product. Install the software product according to the software installation strategy. Document the events and results. ENG.11.BP6: Confirm product readiness. Assure that the software product is ready for use in its intended environment. NOTE 2: Software installation process is linked to Product acceptance support process (SPL.3). ENG.11.BP1: Develop installation strategy. Develop a software installation strategy to install the software product in the target environment in agreement with the customer and the operating organization. NOTE 1: An important part of developing an installation strategy is to develop a strategy to return to the last working system version. In order to be able to re-install the last working version, a complete backup of the system should be made before starting the installation. ENG.11.BP2: Establish installation criteria. Based on the installation requirements, develop criteria for the environment where the software will be installed.
T05. Monitorar e controlar o projeto	
CMMI-DEV	ISO/IEC 15504-5
PMC - SP 1.1 Monitor Project Planning Parameters PMC - SP 1.2 Monitor Commitments PMC - SP 1.3 Monitor Project Risks PMC - SP 1.4 Monitor Data Management PMC - SP 1.5 Monitor Stakeholder Involvement PMC - SP 1.6 Conduct Progress Reviews PMC - SP 2.1 Analyze Issues PMC - SP 2.2 Take Corrective Action PMC - SP 2.3 Manage Corrective Action	MAN.3.BP12: Monitor project attributes. Monitor project scope, budget, cost, resources and other necessary attributes and document significant deviations of them against the project baseline. MAN.3.BP13: Review progress of the project. Regularly report and review the status of the project performance against the project plan. MAN.3.BP14: Act to correct deviations. Take action when project goals are not achieved, to correct deviations from the plan and to prevent recurrence of problems identified in the project. Update project plans accordingly.
T06. Monitorar e Controlar questões financeiras	
CMMI-DEV	ISO/IEC 15504-5
PMC - SP 1.1 Monitor Project Planning Parameters PMC - SP 1.2 Monitor Commitments PMC - SP 1.3 Monitor Project Risks	MAN.3.BP12: Monitor project attributes. Monitor project scope, budget, cost, resources and other necessary attributes and document significant deviations of them against the project baseline.

QUADRO A5.7 – MAPEAMENTO DAS ATIVIDADES DA FASE DE TRANSIÇÃO (CONT.)

T07. Conduzir revisão do milestone: Entrega do Produto	
CMMI-DEV	ISO/IEC 15504-5
PMC - SP 1.7 Conduct Milestone Reviews PP - SP 3.1 Review Plans That Affect the Project PPQA - SP 1.1 Objectively Evaluate Processes PPQA - SP 1.2 Objectively Evaluate Work Products and Services PPQA - SP 2.1 Communicate and Ensure Resolution of Noncompliance Issues PPQA - SP 2.2 Establish Records MA - SP 2.1 Collect Measurement Data MA - SP 2.2 Analyze Measurement Data MA - SP 2.3 Store Data and Results MA - SP 2.4 Communicate Results	SUP.1.BP2: Define quality records. Quality records are defined that demonstrate conformance of process and work products to their quality requirements. SUP.1.BP3: Assure the quality of project process activities and project work products. Carry out a series of activities to provide assurance, with the required level of confidence, that the project processes have followed specified standards and that the work products meet the quality requirements. SUP.1.BP4: Identify and record problems and non-conformances. Problems and nonconformances are identified and recorded and then reported to appropriate stakeholders for information and action. SUP.1.BP5: Act on non-conformances. Deviations or non-conformance with agreed requirements or organizational quality goals are analyzed and resolved. MAN.6.BP5: Collect and store measurement data. Identify, collect and store measurement data, including context information necessary to verify, understand, or evaluate the data. MAN.6.BP6: Analyze measurement data. Analyze and interpret measurement data, and develop information products. MAN.6.BP7: Use measurement information products for decision-making. Make accurate and current measurement information products accessible for any decision-making processes for which it is relevant. MAN.6.BP8: Communicate measurement results. Disseminate measurement information products to all parties who will be using them and collect feedback to evaluate the appropriateness for intended use. MAN.6.BP9: Evaluate and communicate information products and measurement activities to process owners. Evaluate information products and measurement activities against the identified information needs and measurement strategy, identify potential improvements in measurements, and communicate any identified potential improvement to the process owners.

Apêndice 6 – Aplicação do ProcSoftVD – Melhoria

Ramo de atividade da empresa: Desenvolvimento de Sistemas de Informática

Nicho: Comércio Varejista, oficinas mecânicas.

Quadro de funcionários: 3

Fase 1. Diagnóstico do Processo de Software Atual e Definição do Perfil-Alvo

1.1 Descrição do Processo Atual

- É efetuada uma análise inicial, com uma entrevista e anotações (em papel) do que o sistema fará.
- Durante uma semana (mais ou menos) é feita uma análise de todo o sistema, quais ferramentas serão usadas, e é fornecido ao cliente um prazo.
- Durante o processo de desenvolvimento, são efetuadas anotações em papéis e planilhas eletrônicas.
- É visto o que é mais prioritário e este incremento é desenvolvido e entregue. O próprio programador efetua os testes, também. Esses testes são relacionados à compartilhamento de recursos, carga de dados, impressões, e em relação ao software sem utilizar critérios de teste (erros de codificação).
- É efetuado um instalador para a versão, é marcado em uma planilha o número da versão e o que foi instalado ou não.
- Depois, realiza-se o próximo incremento e se refaz o processo novamente

- Nesse ínterim, existem modificações de outros sistemas.

- Um dos funcionários é responsável pelo relacionamento com o cliente, verificando qual será a mudança e se está sendo atendida.
- O outro programador auxilia em testes e alterações.

Quanto ao controle de pagamentos, existe uma planilha para controlar quem pagou o quê em cada mês, pois o produto executável é disponibilizado para o cliente por intermédio de mensalidades.

1.2 Objetivos Estratégicos do Negócio da Empresa

Objetivos Estratégicos:

- Hoje os clientes da empresa são pequenas empresas e um dos objetivos estratégicos é buscar e atender clientes de porte médio e grande, por meio do desenvolvimento de um sistema ERP
- Ter clientes de outras regiões

Metas de melhoria:

- Formalizar e padronizar o processo para desenvolvimento do software (necessária e importante)
- Melhorar a implantação no cliente, deixar explícito para o patrocinador, todas as modificações relevantes
- Desenvolver uma ferramenta de BI (*business intelligence*) que seja integrada ao sistema ERP para dar melhores resultados financeiros.
- Ter meios de suporte *on-line* para o cliente

1.3 Definição dos Perfis-Alvo do Processo

Comercialização

Sintomas da não execução do processo:

- O cliente faz solicitações de mudanças do sistema e pode cancelar a utilização.
- O cliente pode solicitar mais alterações, além do combinado verbalmente, extrapolando prazos, etc

Benefícios:

- O contrato pode estipular multas em casos de solicitações de mudanças e desistência da utilização do sistema, por ex, e isso traz segurança para a empresa desenvolvedora.
- Ter controle das modificações solicitadas pelo cliente dentro um prazo estabelecido.

Modelagem de Negócio

Sintomas da não execução do processo:

- Cada vez que tiver um novo funcionário dentro da *softwarehouse*, alguém terá que ser interrompido para explicar o modelo de negócio.

Benefícios:

- Auxiliar na visão geral dos *stakeholders* do que se trata o negócio do cliente.

Produção de Requisitos

Sintomas da não execução do processo:

- Entendimento único dos processos do sistema (fica “na cabeça” de um participante do projeto)
- A não definição dos requisitos propicia até a inviabilidade do projeto, pois não é avaliado a sua dimensão.

Benefícios:

- Melhor entendimento das lógicas dos processos
- Visão macro de todo o sistema
- A definição dos requisitos ajuda na melhora também do sistema no futuro, pois tem se “um alicerce” de como realmente o sistema funciona

Projeto, Codificação & Integração de Produto

Sintomas da não execução do processo:

- A falta de parâmetros de definição de codificação e integração do produto propicia uma instabilidade no projeto como um todo

Benefícios

- A definição em design, do software, podem levar a uma melhor verificação se o projeto como um todo está funcional ou não.

V&V

Sintomas da não execução do processo:

- Podem surgir diversos erros, ainda mais, no cliente, proporcionando uma corrida pra “atender e resolver” os erros do cliente

Benefícios

- Maior “credibilidade” na versão que está no cliente

Garantia da Qualidade

Sintomas da não execução do processo:

- A falta de avaliação do processo (de forma imparcial) pode proporcionar comodismo no projeto como um todo ou em partes

Benefícios

- Apuração dos resultados, e é possível verificar se está adiantando mesmo fazer todos os processos

Implantação

Sintomas da não execução do processo:

- Dúvidas e falta de credibilidade no ambiente do cliente

Benefícios

- Um controle da implantação propicia maior facilidade da instalação com o cliente

Aquisição

Sintomas da não execução do processo:

- Com as mudanças no software do cliente, uma falta de controle de aquisição, pode gerar erros na instalação de uma versão. (terceirização de parte do software)

Benefícios

- Controlando a aquisição de versões, a *softwarehouse* tem a noção do que está “rodando”, em termos de sistema no cliente.

Medição

Sintomas da não execução do processo:

- A falta de medição pode levar a uma insatisfação do cliente, até porque não é avaliado como o cliente está se adaptando ao sistema

Benefícios

- Consegue-se “medir” um retorno do próprio produto de como ele está, e há várias formas de se medir ele. Ex: em categorias, por índice, etc

Gestão de Requisitos

Sintomas da não execução do processo:

- pode-se gerar uma falta de atualização entre requisitos e o software

Benefícios

- identificando as inconsistências pode evitar-se um desentendimento entre documentação do que é realmente o sistema (em relação a outras pessoas)

Gestão de Projetos

Sintomas da não execução do processo:

- a falta de definição de um plano no projeto como um todo, pode propiciar a que o projeto não chegue a lugar nenhum

Benefícios

- a definição do projeto como um todo, gera a idéia de começo, meio e fim do todo

Gestão de Mudanças e Configuração

Sintomas da não execução do processo:

- falta de parametrização no sistema dos produtos usados, como um todo, em relação a equipe de desenvolvimento

Benefícios

- controle mais apurado de como desenvolver e uso do produto

Gestão de Conhecimento

Sintomas da não execução do processo:

- cada um do desenvolvimento (programadores, analistas) faz e refazem funções, procedimentos, etc, e se perde tempo no desenvolvimento e no projeto

Benefícios

- acho que uma equipe em constante conversação, evita-se e tem-se idéias de inovações no produto e resolução de problemas nos clientes

Fase 2. Análise Estratégica e Planejamento da Mudança

2.1 Priorização das Áreas de Conhecimento

Importância	alta	<ul style="list-style-type: none"> •Gestão de Projetos •Gestão de Requisitos 	<ul style="list-style-type: none"> • V&V •Medição •Aquisição •Gestão de Mudanças e Configuração 	<ul style="list-style-type: none"> • Modelagem de Negócio • Produção de Requisitos
	média	<ul style="list-style-type: none"> •Garantia da Qualidade •Implantação 	<ul style="list-style-type: none"> •Comercialização 	
	baixa	<ul style="list-style-type: none"> •Projeto, Codificação & Integração •Gestão de Conhecimento 		
		baixo	médio	alto

Risco

Ciclos de Melhoria

Ciclo de Melhoria 1: Modelagem de Negócios; Produção de Requisitos

Ciclo de Melhoria 2: V&V

Ciclo de Melhoria 3: Aquisição; Gestão de Mudanças e Configuração

Ciclo de Melhoria 4: Medição

Ciclo de Melhoria 5: Gestão de Requisitos; Gestão de Projetos

Ciclo de Melhoria 6: Garantia da Qualidade; Implantação

Ciclo de Melhoria 7: Comercialização

Ciclo de Melhoria 8: Projeto, Codificação & Integração; Gestão de Conhecimento

2.2 Plano de Ação

Foi definido um patrocinador na empresa, o escopo do projeto de mudança, a WBS, cronograma e avaliados os riscos do projeto de mudança.

Fase 3. Definição das atividades da(s) área(s) de conhecimento

Nessa fase foi realizada a definição das atividades das áreas de conhecimento “Modelagem de Negócios” e “Produção de Requisitos”, tendo como subsídio o Modelo ProcSoftVD.

Fase 4. Implantação da(s) área(s) de conhecimento

Nessa fase as áreas de conhecimento “Modelagem de Negócios” e “Produção de Requisitos” foram institucionalizadas e avaliadas.

Apêndice 7 – Questionário de Avaliação do ProcSoftVD

QUESTIONÁRIO

1. Identificação

1.2 Seu nome (opcional):

1.2 E-mail pessoal para contato:

1.3 Qual é a sua formação acadêmica?

R:

1.4 Função que você exerce na empresa onde trabalha:

a. () Gerente/Supervisor

b. () Gerente de Projetos

c. () Analista de Sistemas

d. () Programador

e. () Outra. Qual ? Nesse caso, algum dia já exerceu a função de analista de sistema e/ou programador em alguma empresa? R:

2. Validação do Modelo ProcSoftVD

2.1 Apresentação do Modelo

Acesse o site <http://www.numa.org.br/GProcSoftVD>.

2.2 Validação das Áreas de Conhecimento

Para cada área de conhecimento será apresentada, a seguir, uma tabela com a sua descrição e espera-se que você defina os pontos fortes e fracos do modelo relacionado a essa área de conhecimento, após sua validação.

DESCRIÇÃO DA ÁREA DE CONHECIMENTO	PONTOS FORTES	PONTOS FRACOS
<p>Comercialização: essa área de conhecimento foi criada para abordar (1) atividades relacionadas à prospecção de potenciais clientes para a empresa (o que não é coberto pelo CMMI, nem pela ISO/IEC 15504-5); (2) atividades relacionadas à elaboração/submissão de propostas ao cliente e negociação e aprovação de contrato sem ambigüidades que especifique as expectativas, responsabilidades, entregas e compromissos de ambas as partes cliente e fornecedor; (3) atividades relacionadas à viabilidade financeira do projeto e relacionadas ao pagamento dos produtos/serviços vendidos pela empresa. As atividades do item (2) foram definidas com subsídio dos processos “Prospecção do Fornecedor” e “Acordado Contratual” da ISO/IEC 15504-5.</p>	•	•
<p>Modelagem de Negócio: o objetivo dessa área de conhecimento é documentar os processos de negócio usando casos de uso de negócio, a fim de assegurar um entendimento comum entre todos os <i>stakeholders</i> (envolvidos) sobre as necessidades existentes no processo de negócio da organização cliente. Essa área de conhecimento teve sua origem no <i>Unified Process</i>.</p>	•	•
<p>Produção de Requisitos: tem o objetivo de produzir e analisar os requisitos do cliente, do produto e dos componentes de produto. Essa área de conhecimento teve sua origem relacionada à área de processo “Desenvolvimento de Requisitos” do CMMI-DEV e aos processos “Elicitação de Requisitos”, “Análise de Requisitos do Sistema” e “Análise de Requisitos do Software” da ISO/IEC 15504-5.</p>	•	•
<p>Projeto, Codificação & Integração de Produto: Engloba os processos “Projeto do Software”, “Integração do Software” e “Integração do Sistema” da 15504-5, o workflow “Implementação” do UP e as áreas de processo “Solução Técnica” e “Integração de Produto” do CMMI-DEV. Segundo a 15504-5, o processo “Projeto de Software” tem o objetivo de fornecer um design para o software que é implementado e pode ser verificado em confronto aos requisitos; o processo “Integração de Software” tem o objetivo de combinar as unidades de software, produzindo itens de software integrados, consistentes com o projeto de software, os quais demonstram que os requisitos funcionais e não-funcionais foram satisfeitos; e o processo “Integração do Sistema” tem como objetivo integrar os elementos do sistema (incluindo os itens de software, itens de hardware, operações manuais e outros sistemas, se necessário) para produzir um sistema completo que satisfaça ao projeto do sistema e às expectativas do cliente, expressadas em requisitos do sistema. Segundo o modelo CMMI-DEV, a área de processo “Solução Técnica” tem como objetivo projetar, desenvolver e implementar soluções para os requisitos, soluções essas que envolvem produtos, componentes de produtos e processos de ciclo de vida relacionados ao produto; e a área de processo “Integração do Produto” tem o objetivo de montar o produto, a partir dos componentes de produto, assegurar que o produto ao ser integrado funcione adequadamente, e entregar o produto.</p>	•	•
<p>V&V: Engloba as áreas de processo (CMMI) e processos</p>	•	•

<p>(15504-5) “Validação” e “Verificação” e os processos “Teste de Software” e “Teste de Sistema” da 15504-5. O objetivo da área de processo/processo “Validação” é demonstrar que o produto ou componente do produto atenda ao uso pretendido quando colocado no ambiente destinado. Já o objetivo da área de processo/processo “Verificação” é assegurar que produtos de trabalho (e serviços, no caso do CMMI) selecionados alcancem seus requisitos especificados. Testes desempenham um papel extremamente importante em V&V (Verificação e Validação). Segundo o modelo 15504-5, o processo “Teste” pode ser realizado tanto no software quanto no sistema. O processo “Teste de Software” tem como objetivo confirmar que o produto de software integrado atende aos requisitos definidos. E, o processo “Teste de Sistema” tem o objetivo de assegurar que a implementação de cada requisito de sistema seja testada quanto à sua conformidade e que o sistema esteja pronto para entrega.</p>		
<p>Implantação: o objetivo é entregar o produto produzido para os usuários finais, por meio das atividades de codificação, teste, empacotamento e instalação do software. Essa área de conhecimento teve sua origem no UP e também está relacionada ao processo “Entrega de Produto” da 15504-5.</p>	•	•
<p>Aquisição: essa área de conhecimento tem o objetivo de gerenciar a aquisição de produtos de fornecedores, sejam equipamentos ou até mesmo componentes de software do produto (no caso de terceirização do serviço). Exemplos de produtos e componentes de produtos que podem ser adquiridos pelo projeto: subsistemas (por exemplo, um sistema navegacional de uma aeronave), software, hardware, documentação (como manuais de instalação, de operação e do usuário). A origem dessa área de conhecimento está relacionada ao grupo de processo “Aquisição” da 15504-5 à área de processo denominada “Gestão de Acordo com Fornecedor” do CMMI-DEV.</p>	•	•
<p>Medição: essa área de conhecimento tem o objetivo de coletar e analisar dados relacionados aos produtos desenvolvidos e aos processos implementados dentro da organização por meio de projetos, a fim de dar um suporte efetivo à gestão dos processos e demonstrar objetivamente a qualidade dos produtos. Sua origem está relacionada à área de processo “Análise e Medição” do CMMI-DEV e do processo “Medição” da 15504-5.</p>	•	•
<p>Garantia da Qualidade de Produto e Processo: o objetivo dessa área de conhecimento é fornecer a garantia de que processos e produtos de trabalho estejam em conformidade com planos e provisões pré-definidos. Sua origem está relacionada tanto ao CMMI quanto à ISO/IEC 15504.</p>	•	•
<p>Gestão de Requisitos: o objetivo dessa área de conhecimento, que teve sua origem no CMMI-DEV, é gerenciar os requisitos dos produtos e componentes de produtos dos projetos e identificar inconsistências entre esses requisitos e os planos e produtos de trabalho dos projetos. Para isso, essa área trata do rastreamento dos requisitos em meio ao projeto e das mudanças desses requisitos. As mudanças relacionadas aos requisitos utilizam algumas das atividades definidas pela área de conhecimento “Gestão de Mudanças e Configuração”.</p>	•	•
<p>Gestão de Mudanças e Configuração: essa área de conhecimento engloba a área de processo “Gestão de</p>	•	•

Configuração” do CMMI-DEV e os processos “Gestão de Configuração” e “Gestão de Solicitação de Mudanças” da 15504-5. O objetivo dessa área de conhecimento é estabelecer e manter a integridade de produtos de trabalho usando a identificação de configuração, controle de configuração, prestação de contas (explicação) do status da configuração e auditoria da configuração, além de assegurar que solicitações de mudanças no projeto sejam gerenciadas, rastreadas e controladas.		
Gestão de Projetos: o objetivo dessa área de processo é identificar, estabelecer, coordenar e monitorar as atividades, tarefas e recursos necessários para um projeto produzir um produto e/ou serviço, no contexto dos requisitos e restrições de projetos. Engloba tanto o processo “Gestão de Projetos” da 15504-5 quanto às áreas de processo “Planejamento de Projeto” e “Monitoramento e Controle de Projeto” do CMMI.	•	•
Gestão de Conhecimento: tem como objetivo assegurar que o conhecimento, informação e habilidades individuais sejam coletadas, compartilhadas, reusadas e melhoradas por toda a organização. Sua origem está relacionada ao processo “Gestão de Conhecimento” da ISO/IEC 15504.	•	•

2.3 Validação do Modelo como um todo

2.3.1 Por gentileza, abra o arquivo “CMMI-v1-2.pdf”. Aperte SHIFT+CTRL+N e digite 339. Você foi redirecionado para a área de processo do CMMI “Planejamento de Projeto”. Nela você encontra o objetivo dessa área, os objetivos específicos e para cada um deles as sub-práticas e produtos de trabalho típicos. Todas as outras áreas de processo têm a mesma estrutura.

- Para uma MPE, o que é mais fácil: utilizar como modelo de referência para criar o seu processo padrão o CMMI ou um outro modelo com mais detalhes de como podem ser realizadas as atividades do processo, como o proposto nesse trabalho de pesquisa?

R: _____

2.3.2 Por gentileza, abra o arquivo “ISO-IEC_FDIS_15504-5.pdf”. Aperte SHIFT+CTRL+N e digite 69. Você foi redirecionado para o processo “Gestão de Projeto”. Nele você encontra o objetivo desse processo, os resultados, as práticas-base e os produtos de trabalho típicos. Todos os outros processos têm a mesma estrutura.

- Para uma MPE, o que é mais fácil: utilizar como modelo de referência para criar o seu processo padrão a norma ISO/IEC 15504-5 ou um outro modelo com mais detalhes de como podem ser realizadas as atividades do processo, como o proposto nesse trabalho de pesquisa?

R: _____

2.3.3 Você enxerga o modelo proposto nesse trabalho de pesquisa como uma referência, um guia que poderia ser utilizado por uma MPE para, a partir dele, definir o seu processo padrão (mesmo que seja necessário fazer alguns ajustes para adequar à realidade da empresa)?

R: _____

2.3.4 De forma geral, quais os comentários você faria a respeito do modelo proposto?

R: _____

2.3.5 Acesse a planilha “AvaliacaoQualidade.xls” e preencha a coluna “Qual nível de atendimento de cada sub-característica?” com as alternativas do combo-box (Alto, Médio, Baixo e NA (não se aplica)). Por favor, envie a planilha preenchida junto a esse questionário.

Apêndice 8 – Pontos Fortes e Fracos Transcritos das Respostas do Questionário entregue aos Profissionais

As respostas dos 6 (seis) profissionais que avaliaram o modelo quanto aos seus pontos fortes e fracos estão transcritas, por área de conhecimento, nos Quadros A8.1 a A8.13.

QUADRO A8.1 – TRANSCRIÇÃO DAS RESPOSTAS – ÁREA DE CONHECIMENTO PRODUÇÃO DE REQUISITOS

Quanto à Produção de Requisitos	Pontos Fortes	P1: "O <i>template</i> documento de requisitos é simplificado e fácil de ser colocado em prática; muito bom".
		P2: "a) Documentação muito bem definida; b) Contempla Exemplos/Sugestões; c) Completo".
		P3: não comentou
		P4: não comentou
		P5: "Tempo despendido em modelagem, análise e produção de requisitos é um excelente investimento, e não um gasto".
		P6: "a) Vasto material de apoio para a elicitação dos requisitos para o software; b) Bem detalhado".
	Pontos Fracos	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: não comentou
		P5: "Muito material para ser lido, analisado, documentado e empreendido, para autônomos e equipes pequenas".
		P6: não comentou

QUADRO A8.2 – TRANSCRIÇÃO DAS RESPOSTAS – ÁREA DE CONHECIMENTO MODELAGEM DE NEGÓCIOS

Quanto à Modelagem de Negócios	Pontos Fortes	P1: "As tarefas que incluem a modelagem de negócio estão claramente definidas".
		P2: "a) Completo e eficiente; b) Foco e preocupação com aspectos fundamentais dessa área de conhecimento; c) Clareza nas explicações; d) Contempla Exemplos/Sugestões".
		P3: não comentou
		P4: não comentou
		P5: não comentou
		P6: "Roteiro bem definido, enfatizando a importância da interação entre cliente e desenvolvedor".
	Pontos Fracos	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: não comentou
		P5: não comentou
		P6: não comentou

QUADRO A8.3 – TRANSCRIÇÃO DAS RESPOSTAS – ÁREA DE CONHECIMENTO COMERCIALIZAÇÃO

Quanto à Comercialização	Pontos Fortes	P1: “As atividades prospectar clientes, visitar clientes e criar infra-estrutura do projeto são pontos fortes do modelo”.
		P2: “a) Fornece plenas condições para o gerenciamento do contato inicial com o cliente, transmitindo eficiência e credibilidade perante o cliente; b) Aspecto formal abrangido sem exageros; ideal!; c) Contempla Exemplos/Sugestões”.
		P3: não comentou
		P4: “Cobertura de uma atividade cuja organização desenvolvidora deve se preocupar”.
		P5: “a) O modelo contempla a viabilidade do projeto para a empresa; b) Com o auxílio do Marketing pode-se aumentar o desenvolvimento de produtos, aumentando-se os lucros”.
		P6: “a) Linguagem de fácil entendimento por parte da empresa envolvida; b) Roteiro bem simplificado, a área de conhecimento é bem explicada e fácil de se seguir; c) Elaboração do contrato de prestação de serviço junto ao cliente ou de venda do software bem explicado e de acordo com a legislação”.
	Pontos Fracos	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: “É preciso considerar dois papéis claramente: cliente (consumidor de uma solução computacional) e fornecedor (produtor dessa solução). A análise da viabilidade financeira é uma atividade do cliente, não do fornecedor. Aliás, o cliente já tem parâmetros para tal análise, o que torna essa atividade sem uma utilidade para ele. E para o fornecedor? Por que uma MPE deveria usar essa atividade?”
		P5: “No pós-venda a equipe de vendas não acompanha o andamento do cliente, sendo que é necessário um acompanhamento mútuo de suporte”.
		P6: não comentou

QUADRO A8.4 – TRANSCRIÇÃO DAS RESPOSTAS – ÁREA DE CONHECIMENTO PROJETO, CODIFICAÇÃO & INTEGRAÇÃO DE PRODUTO

Quanto ao Projeto, Codificação & Integração de Produto	Pontos Fortes	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: não comentou
		P5: “Um software deve ser bem projetado, desenvolvido e documentado de forma a permitir uma fácil manutenção e evolução do mesmo. Os custos de manutenção são muito superiores ao custo de desenvolvimento do sistema. O modelo apresenta atividades e <i>templates</i> para a realização dessa área de conhecimento que podem auxiliar”.
		P6: “a) Bem detalhado; b) Fornece subsídios para o início da codificação por parte dos responsáveis pelo desenvolvimento do software”.
	Pontos Fracos	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: não comentou
		P5: não comentou
P6: não comentou		

QUADRO A8.5 – TRANSCRIÇÃO DAS RESPOSTAS – ÁREA DE CONHECIMENTO AQUISIÇÃO

Quanto à Aquisição	Pontos Fortes	P1: não comentou
		P2: "Clareza".
		P3: não comentou
		P4: "Parece coerente"
		P5: "Fazer cotações e análise de fornecedores é importante para viabilizar o projeto".
		P6: "a) Auxilia o desenvolver no gerenciamento de aquisição de hardwares, por exemplo, que estarão sendo usados para a implantação do software; b) Vasto material de apoio; c) Bem detalhado".
	Pontos Fracos	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: não comentou
		P5: "Sugiro inserir o Documento de requisitos do componente adquirido pelo fornecedor, no caso de componente de software, como artefato do projeto que poderá ser modificado e, por isso, precisa ser colocado sob gerência de requisitos também".
		P6: não comentou

QUADRO A8.6 – TRANSCRIÇÃO DAS RESPOSTAS – ÁREA DE CONHECIMENTO MEDIÇÃO

Quanto à Medição	Pontos Fortes	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: "Parece coerente"
		P5: "É importantíssimo avaliar os processos como forma de aprimoramento e diminuição de redundâncias nos processos".
		P6: "O modelo fornece subsídios para se ter dados relacionados a todos os projetos da empresa; dados como tempo estimado e concretizado, valor cobrado, etc, que serão utilizados para estimativas em projetos futuros".
	Pontos Fracos	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: não comentou
		P5: não comentou
		P6: não comentou

QUADRO A8.7 – TRANSCRIÇÃO DAS RESPOSTAS – ÁREA DE CONHECIMENTO IMPLANTAÇÃO

Quanto à Implantação	Pontos Fortes	P1: não comentou
		P2: "a) Preocupação com a satisfação do cliente; b) Bem gerenciado".
		P3: não comentou
		P4: "Parece coerente"
		P5: "O modelo apresenta atividades relacionadas à implantação dos <i>releases</i> do produto ao cliente de forma completa".
		P6: "Cuida de todos os tópicos relacionados à implantação e a entrega do material."
	Pontos Fracos	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: não comentou
		P5: não comentou
		P6: não comentou

QUADRO A8.8 – TRANSCRIÇÃO DAS RESPOSTAS – ÁREA DE CONHECIMENTO V&V

Quanto à V&V	Pontos Fortes	P1: não comentou
		P2: “a) Muito bem distribuído e especificado nas fases; b) Presença marcante no modelo proposto”.
		P3: não comentou
		P4: não comentou
		P5: “Teste é uma das tarefas mais importantes e que não deve ser negligenciada durante o desenvolvimento, com apenas testes rápidos e sem critérios”.
		P6: “Muito importante, porque nem sempre as empresas de software testam e validam seus produtos antes de entregar para o cliente”.
	Pontos Fracos	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: não comentou
		P5: não comentou
		P6: não comentou

QUADRO A8.9 – TRANSCRIÇÃO DAS RESPOSTAS – ÁREA DE CONHECIMENTO GARANTIA DA QUALIDADE DE PRODUTO E PROCESSO

Quanto à Garantia da Qualidade de Produto e Processo	Pontos Fortes	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: “Parece coerente”
		P5: “A verificação de qualidade é um processo apontado aqui em todas as etapas de desenvolvimento”.
		P6: não comentou
	Pontos Fracos	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: não comentou
		P5: não comentou
		P6: não comentou

QUADRO A8.10 – TRANSCRIÇÃO DAS RESPOSTAS – ÁREA DE CONHECIMENTO GESTÃO DE CONHECIMENTO

Quanto à Gestão de Conhecimento	Pontos Fortes	P1: não comentou
		P2: “Garantia de controle e disponibilidade do conhecimento”.
		P3: não comentou
		P4: não comentou
		P5: não comentou
		P6: “O modelo mostra uma forma de gerenciar o conhecimento de todos os desenvolvedores da empresa, fazendo com que o conhecimento e experiência de um desenvolvedor sejam disseminados para outros interessados”.
	Pontos Fracos	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: não comentou
		P5: não comentou
		P6: não comentou

QUADRO A8.11 – TRANSCRIÇÃO DAS RESPOSTAS – ÁREA DE CONHECIMENTO GESTÃO DE REQUISITOS

Quanto à Gestão de Requisitos	Pontos Fortes	P1: não comentou
		P2: "Muito bem articulado e completo".
		P3: não comentou
		P4: não comentou
		P5: "Importante para controlar qualquer tipo de requisito que possa influenciar na lógica de processos".
		P6: "O gerenciamento de todos os requisitos é muito importante e é contemplado pelo modelo".
	Pontos Fracos	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: não comentou
		P5: "Sugiro inserir o Documento de requisitos do componente adquirido pelo fornecedor, no caso de componente de software, como artefato do projeto que poderá ser modificado e, por isso, precisa ser colocado sob gerência de requisitos também".
		P6: não comentou

QUADRO A8.12 – TRANSCRIÇÃO DAS RESPOSTAS – ÁREA DE CONHECIMENTO GESTÃO DE PROJETOS

Quanto à Gestão de Projetos	Pontos Fortes	P1: não comentou
		P2: "Muito bem articulado e completo".
		P3: não comentou
		P4: não comentou
		P5: "É importante entender o escopo do projeto e fazer o planejamento preliminar desse escopo, a fim de se ter uma visão macro em relação ao que se pretende atingir com o produto resultante do projeto, quanto será gasto, quanto demorará para ser produzido e qual o retorno financeiro desse projeto. E, também, é importante fazer o monitoramento e controle desse projeto para que se possa tomar medidas reativas e até mesmo pró-ativas. Esse modelo atende a esses requisitos".
		P6: "O Gerenciamento de Projetos é muito importante e é contemplado pelo modelo".
	Pontos Fracos	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: não comentou
		P5: não comentou
		P6: não comentou

QUADRO A8.13 – TRANSCRIÇÃO DAS RESPOSTAS – ÁREA DE CONHECIMENTO GESTÃO DE MUDANÇAS E CONFIGURAÇÃO

Quanto à Gestão de Mudanças e Configuração	Pontos Fortes	P1: não comentou
		P2: "Completo e detalhado".
		P3: não comentou
		P4: não comentou
		P5: "Implementações e mudanças em sistemas são processos constantes e, portanto, é necessário um controle dessas modificações".
		P6: "O gerenciamento das versões dos softwares desenvolvidos pela empresa é muito importante e o modelo fornece sugestões de como fazê-lo".
	Pontos Fracos	P1: não comentou
		P2: não comentou
		P3: não comentou
		P4: não comentou
		P5: "Acho importante um Controle para Help-Desk"
		P6: não comentou

Apêndice 9 – Respostas do Questionário respondido pelos Profissionais

Os Quadros A9.1 a A9.4 apresentam as respostas transcritas relacionadas a 4 (quatro) questões respondidas pelos 6 (seis) profissionais que avaliaram o modelo.

QUADRO A9.1 – TRANSCRIÇÃO DA RESPOSTA PARA A QUESTÃO 1

Questão 1	Respostas
Para uma MPE, o que é mais fácil: utilizar como modelo de referência para criar o seu processo padrão o CMMI ou um outro modelo com mais detalhes de como podem ser realizadas as atividades do processo, como o proposto nesse trabalho de pesquisa?	P1: “Um outro modelo com mais detalhes”.
	P2: “O melhor é utilizar um modelo como o proposto neste trabalho, pois uma MPE não teria recursos financeiros, pessoas e tempo para criar um modelo com base no CMMI. Além disso, os detalhes e as orientações de como e o que deve ser feito auxiliam muito. Outro fator é a clareza e a riqueza de detalhes que tornam a adoção do modelo um processo mais rápido e eficiente”.
	P3: “A utilização do modelo apresentado nesse trabalho torna-se mais efetivo que o CMMI, uma vez que sugere os <i>templates</i> e documentos de apoio. A aplicação de qualquer modelo que cita diretrizes gerais dificulta a aplicação, pois deixa sem direcionamento o procedimento de aplicação”.
	P4: “Acredito que para uma MPE seria mais fácil iniciar com um modelo simplificado, e não com um modelo com mais atividades. A estrutura organizacional é geralmente pequena para tantas atividades. Dessa forma, à medida que a organização cresce, tem condições de evoluir seu processo de desenvolvimento e comercialização”.
	P5: “Na minha opinião, o modelo CMMI é muito complexo, e acho que o modelo apresentado nesse trabalho poderia auxiliar mais (logicamente adaptado à realidade da empresa)”.
	P6: “É mais fácil utilizar um modelo com mais detalhes porque cada atividade de cada área de conhecimento foi detalhada, fornecendo material suficiente para que cada empresa utilize os modelos do jeito que lhe for mais útil”.

QUADRO A9.2 – TRANSCRIÇÃO DA RESPOSTA PARA A QUESTÃO 2

Questão 2	Respostas
Para uma MPE, o que é mais fácil: utilizar como modelo de referência para criar o seu processo padrão a norma ISO/IEC 15504-5 ou um outro modelo com mais detalhes de como podem ser realizadas as atividades do processo, como o proposto nesse trabalho de pesquisa?	P1: “Um outro modelo com mais detalhes”.
	P2: “Um outro modelo com mais detalhes”.
	P3: “Apesar da norma ISO/IEC 15504-5 ter mais detalhes com relação aos procedimentos, ainda não contém os <i>templates</i> que podem ser o diferencial na hora de uma aplicação real”.
	P4: “Acredito que para uma MPE seria mais fácil iniciar com um modelo simplificado, e não com um modelo com mais atividades. A estrutura organizacional é geralmente pequena para tantas atividades. Dessa forma, à medida que a organização cresce, tem condições de evoluir seu processo de desenvolvimento e comercialização”.
	P5: “Na minha opinião, comparando o CMMI com a 15504-5, acredito que o último é muito mais fácil de ser compreendido e com possibilidade de ser usado até mesmo por uma MPE como subsídio para criar o seu processo padrão. Entretanto, apesar de apresentar práticas base, resultados, produtos de trabalho típicos e até mesmo sugestões de itens a serem abordados nesses produtos de trabalho, a ISO 15504-5 (assim como o CMMI) não fornece templates mais detalhados a respeito desses produtos, o que facilitaria e muito o entendimento de “como fazer” para realizar as atividades em MPEs”.
	P6: “Um outro modelo com mais detalhes, como o apresentado nesse trabalho”.

QUADRO A9.3 – TRANSCRIÇÃO DA RESPOSTA PARA A QUESTÃO 3

Questão 3	Respostas
Você enxerga o modelo proposto nesse trabalho de pesquisa como uma referência, um guia que poderia ser utilizado por uma MPE para, a partir dele, definir o seu processo padrão (mesmo que sejam necessários alguns ajustes para adequar à realidade da empresa)?	P1: “Sim”.
	P2: “Sim, pois o modelo aborda todos os aspectos relevantes do processo de desenvolvimento de software para as MPEs, enfocando especificamente suas características e necessidades (busca de clientes, agilidade, inexperiência na área de Engenharia de Software, recursos limitados). É extremamente viável, pois além de descrever cada fase do modelo proposto com um vocabulário de fácil entendimento, ainda indica/sugere formas para sua aplicação na empresa. Além disso, possui referências e anexos contendo a fundamentação teórica sobre cada assunto que pode não ser trivial ao seu público-alvo”.
	P3: “Sim. Acredito que as áreas podem ser instanciadas isoladamente para melhoria de determinadas áreas de um processo, especialmente, em empresas que já detêm algum procedimento organizado de concepção e construção de software. Entretanto, para MPE ainda acredito ser necessário algumas simplificações, pois o modelo está complexo e exige um grau de conhecimento técnico e de recursos humanos que pode inviabilizar seu uso”.
	P4: “Pelo que tenho visto, seria ideal. Mas a realidade é outra. Ele seria útil para conscientizar uma MPE sobre a necessidade e vantagens de se ter um processo padrão, e até mesmo considerar tarefas que podem não ser consideradas em tal organização (até por não ter sentido a necessidade delas)”.
	P5: “Sim. Aliás, se encaixa muito no atual processo organizacional de minha empresa de software, com apenas pequenos ajustes, para adaptação do modelo à realidade da empresa”.
	P6: “Sim. Para a minha empresa que está iniciando suas atividades, onde eu serei responsável por todas as áreas de conhecimento, o modelo será de muito auxílio, pois seguirei normas estabelecidas e estarei de acordo com o que o mercado consumidor na área de TI quer: softwares bem projetados e empresas de software com seus processos de desenvolvimento bem definidos”.

QUADRO A9.4 – TRANSCRIÇÃO DA RESPOSTA PARA A QUESTÃO 4

Questão 4	Respostas
De forma geral, quais os comentários você faria a respeito do modelo proposto?	P1: “Este modelo é um guia de referência espetacular para uma MPE criar seus processos de software. É um modelo genérico que serve para a empresa vender, modelar e executar qualquer tipo de processo de software, sem a necessidade da contratação de consultores especializados neste tipo de trabalho por um tempo muito grande, o que seria inviável para uma MPE. O modelo proposto é fácil de entender, pois contém todas as atividades necessárias para a venda e desenvolvimento de software, descritas de uma maneira simples e completa, com ênfase às tarefas de cada atividade, bem como suas entradas e saídas, quem deve executá-las e os recursos que podem ser utilizados, tornando o processo auto-explicativo e possível de ser conduzido por uma MPE”.
	P2: “Com eficiência e baixo custo, o modelo proporciona o gerenciamento de projetos, auxilia na melhoria da qualidade dos produtos desenvolvidos. O detalhamento das fases e as orientações sobre “o quê” e “como” realizar as tarefas é extremamente útil e relevante, uma vez que facilita o uso e adaptação ao modelo. Outro ponto relevante é a área de conhecimento “comercialização” que foi inserida no modelo e que é fundamental para as empresas de desenvolvimento de software, pois envolve atividades de busca por potenciais clientes, o que muitas vezes as empresas não dedicam o devido tempo. Seria interessante que o modelo fosse testado/experimentado por alguma(s) empresa(s) de desenvolvimento de software. Assim, várias opiniões e sugestões significativas poderiam ser obtidas.”
	P3: “Alguns exemplos estão em inglês e outros em português. Seria interessante definir uma padronização, idealmente, em português. Seria também interessante explicitar em todas as tarefas das atividades, que é possível consultar/atualizar a base de conhecimento da empresa”.
	P4: “A organização está um tanto confusa. Não fica claro como os ciclos do RUP (que é baseado em modelo Espiral) ficariam nesse processo. Também não fica claro como uma organização utilizaria tal processo para o primeiro projeto. Por exemplo, considere uma proposta de uma empresa a ser incubada em uma Incubadora Tecnológica, que ainda não tem cliente e não tem produto. Nesse caso, vários artefatos utilizados como subsídios ainda não existem. Como fica o processo? Para melhorar, facilitando seu uso por uma MPE, o uso de um modelo o processo para apoiar a definição de um processo deveria ser evolutivo, iniciando com atividades fundamentais, com vistas ao crescimento e facilidade de evolução do processo simplificado (inicial) para um processo mais completo e maduro, conforme a organização evolui (cresce)”.
	P5: “Achei muito interessante o modelo por se adequar realmente à realidade de uma MPE, uma vez que é importante estudar tendências que o mercado direciona”.
	P6: “Muito bom - o modelo fornece às MPEs uma orientação de como podem realizar as atividades de um processo de venda e desenvolvimento de software, possibilitando que a empresa adapte o modelo às suas necessidades”.

Livros Grátis

(<http://www.livrosgratis.com.br>)

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)
[Baixar livros de Literatura de Cordel](#)
[Baixar livros de Literatura Infantil](#)
[Baixar livros de Matemática](#)
[Baixar livros de Medicina](#)
[Baixar livros de Medicina Veterinária](#)
[Baixar livros de Meio Ambiente](#)
[Baixar livros de Meteorologia](#)
[Baixar Monografias e TCC](#)
[Baixar livros Multidisciplinar](#)
[Baixar livros de Música](#)
[Baixar livros de Psicologia](#)
[Baixar livros de Química](#)
[Baixar livros de Saúde Coletiva](#)
[Baixar livros de Serviço Social](#)
[Baixar livros de Sociologia](#)
[Baixar livros de Teologia](#)
[Baixar livros de Trabalho](#)
[Baixar livros de Turismo](#)