

ANTONIO CARLOS MARQUES DO AMARAL GUERRA

**UMA FERRAMENTA PARA APOIO  
À GESTÃO DE ESCOPO DE PROJETO  
EM TECNOLOGIA DA INFORMAÇÃO**

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Ciências, conferido pela Universidade Federal de Uberlândia - UFU, através da Faculdade de Engenharia Elétrica.

Uberlândia  
2006

# **Livros Grátis**

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

Milhares de livros grátis para download.

ANTONIO CARLOS MARQUES DO AMARAL GUERRA

**UMA FERRAMENTA PARA APOIO  
À GESTÃO DE ESCOPO DE PROJETO  
EM TECNOLOGIA DA INFORMAÇÃO**

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Ciências, conferido pela Universidade Federal de Uberlândia - UFU, através da Faculdade de Engenharia Elétrica.

Banca Examinadora:

Prof. Dra. Selma Shin Shimizu Melnikoff – USP

Prof. PhD. Keiji Yamanaka

Prof. Dr. Alexandre Cardoso – (Orientador)

Prof. PhD. Edgard A. Lamounier Jr. (Co-orientador)

Uberlândia

2006

## FICHA CATALOGRÁFICA

Elaborada pelo Sistema de Bibliotecas da UFU / Setor de Catalogação e Classificação

G934f Guerra, Antonio Carlos Marques do Amaral.  
Uma ferramenta para apoio à gestão de escopo de projeto em tecnologia da informação / Antonio Carlos Marques do Amaral Guerra. - Uberlândia, 2006.  
139f. : il.  
Orientador: Alexandre Cardoso.  
Dissertação (mestrado) – Universidade Federal de Uberlândia, Programa de Pós-Graduação em Engenharia Elétrica.  
Inclui bibliografia.  
1. Engenharia elétrica - Teses. 2. Administração de projetos - Teses.  
3. Software - Desenvolvimento - Teses. I. Cardoso, Alexandre. II. Universidade Federal de Uberlândia. Programa de Pós-Graduação em Engenharia Elétrica. III. Título.

CDU: 621.3

## **Dedicatória**

A Deus, nosso criador e nossa mãe Maria  
Santíssima, pela força, saúde e perseverança  
para a conclusão deste trabalho;

A meus pais, amigos e familiares,  
pela compreensão por muitos momentos que  
estive ausente, dedicando meu tempo e atenção  
na elaboração deste trabalho;

De modo especial, à minha esposa Sandra,  
e meus filhos Leandro e Cristiane.

## **Agradecimentos**

Quero externar gratidão a todos os amigos que de alguma forma colaboraram com a realização deste trabalho;

Aos meus colegas de trabalho, pelo apoio e companheirismo, tanto da Companhia Siderúrgica Paulista, na qual trabalho desde 1985, quanto da Universidade Santa Cecília;

Aos professores da Universidade Federal de Uberlândia, que tive contato durante as aulas deste curso, pelos ensinamentos que transmitiram;

Agradecimento especial aos meus orientadores, Prof. Dr. Alexandre Cardoso e Prof. PhD. Edgard Afonso Lamounier Jr. pela paciência e ensinamentos, que mesmo com o encerramento desta etapa, não serão jamais esquecidos.

## SUMÁRIO

LISTA DE FIGURAS .....	I
LISTA DE TABELAS .....	II
LISTA DE FÓRMULAS .....	III
ABREVIATURAS E SIGLAS .....	IV
RESUMO .....	V
ABSTRACT .....	VI
1 INTRODUÇÃO .....	1
1.1 Motivação .....	1
1.2 Objetivos .....	3
1.3 Estrutura do Trabalho .....	4
2 FUNDAMENTOS .....	5
2.1 Engenharia de Requisitos .....	5
2.1.1 Levantamento de Requisitos .....	5
2.1.1.1 Entrevistas .....	6
2.1.1.2 Questionários .....	7
2.1.1.3 Observação Direta .....	8
2.1.1.4 Pesquisa Bibliográfica .....	8
2.1.1.5 JAD – Joint Application Design .....	9
2.1.2 Análise e Negociação de Requisitos .....	10
2.1.3 Especificação de Requisitos .....	11
2.1.4 Validação de Requisitos .....	12
2.1.5 Gerenciamento de Requisitos .....	12
2.2 Use Case (Caso de Uso) .....	13
2.3 Diagrama de Transição de Estado (DTE) .....	14
2.4 Métricas de Software .....	15
2.4.1 LOC – Lines Of Code (Linhas de Código Fonte) .....	16

2.4.2	APF – Análise de Pontos de Função . . . . .	17
2.4.3	COSMIC-FFP . . . . .	18
2.4.4	NESMA - Nederlandse Software Metrieken Associatie . . . . .	18
2.4.5	UCP – Use Case Points (Pontos por Caso de Uso) . . . . .	19
2.4.6	COCOMO - Cost Constructive Model . . . . .	20
2.4.7	COCOMO II . . . . .	21
2.4.8	Análise Comparativa das Métricas de Software . . . . .	21
2.5	PMI (PROJECT MANAGEMENT INSTITUTE) . . . . .	23
2.5.1	PMI no mundo . . . . .	23
2.5.2	PMI no Brasil . . . . .	23
2.5.3	PMBOK . . . . .	24
2.5.3.1	Áreas de Conhecimento do PMBOK . . . . .	25
2.5.3.2	Grupos de Processos . . . . .	26
2.5.4	Certificação PMP – Project Management Professional . . . . .	26
2.6	CMMI (CAPABILITY MATURITY MODEL INTEGRATION) . . . . .	27
2.6.1	Introdução . . . . .	27
2.6.2	Níveis de Maturidade . . . . .	27
2.6.3	Áreas Chaves de Processos . . . . .	28
2.6.4	Certificação CMMI . . . . .	29
3	TRABALHOS RELACIONADOS . . . . .	30
3.1	Introdução . . . . .	30
3.2	PRAXIS (Processo para Aplicativos Extensíveis Interativos) . . . . .	30
3.2.1	Visão Geral do Praxis . . . . .	30
3.2.2	Análise Comparativa . . . . .	32
3.3	SIGERE (Sistema de Gestão de Requisitos) . . . . .	32
3.3.1	Visão Geral do SIGERE . . . . .	32
3.3.2	Análise Comparativa . . . . .	33
3.4	GEMETRICS (Ferramenta para Gestão de Projetos e Métricas) . . . . .	34
3.4.1	Visão Geral do GEMETRICS . . . . .	34
3.4.2	Análise Comparativa . . . . .	35
4	ARQUITETURA - Sistema de Apoio à Gestão do Escopo do Projeto . . . . .	36



4.1	Introdução . . . . .	36
4.2	Requisitos do Sistema . . . . .	37
4.3	Segurança Lógica e Física do Sistema . . . . .	37
4.4	Funções do SAGEP . . . . .	38
4.4.1	Funções de Apoio ao Sistema . . . . .	38
4.4.2	Funções de Gestão de Solicitações de Mudanças . . . . .	38
4.4.3	Funções de Apoio às estimativas de Prazo e Custo . . . . .	38
4.4.4	Funções de Posicionamento de progresso do projeto . . . . .	38
4.5	Descrição do Sistema . . . . .	39
4.5.1	Início de um projeto . . . . .	39
4.5.2	Cadastramento do Escopo Inicial do projeto . . . . .	41
4.5.3	Necessidade de Mudança . . . . .	41
4.5.4	Apoio ao cálculo de Pontos de Função . . . . .	45
4.5.5	Apoio à Estimativa de Prazo e Custo através do Histórico . . . . .	47
4.5.6	Posicionamento da Situação do projeto . . . . .	49
4.5.7	Consultas Diversas . . . . .	50
4.5.8	Final do projeto . . . . .	51
5	ESTUDO DE CASO . . . . .	52
5.1	Introdução . . . . .	52
5.2	Situação encontrada . . . . .	52
5.3	Desenvolvimento da Ferramenta de Apoio . . . . .	53
5.4	Análise de Resultados . . . . .	55
5.5	Análise da ferramenta SAGEP . . . . .	58
5.6	Considerações Finais . . . . .	59
6	CONCLUSÃO . . . . .	60
6.1	Introdução . . . . .	60
6.2	Conclusões . . . . .	60
6.3	Sugestões para trabalhos futuros . . . . .	61
5.6	Considerações Finais . . . . .	62

ANEXO A - Detalhamento da Análise de Pontos de Função (APF) . . . . .	63
1 Contagem de pontos de função não ajustados (PFNA) . . . . .	64
2 Determinar o valor do fator de ajuste . . . . .	70
2.1 Comunicação de Dados . . . . .	70
2.2 Funções Distribuídas . . . . .	71
2.3 Desempenho da Execução do Sistema . . . . .	71
2.4 Carga de Máquina (Utilização do equipamento) . . . . .	72
2.5 Volume de Transações . . . . .	72
2.6 Entrada de Dados On-Line . . . . .	72
2.7 Eficiência do Usuário Final (Interface com o usuário) . . . . .	73
2.8 Atualizações On-Line . . . . .	73
2.9 Processamento Complexo . . . . .	74
2.10 Reusabilidade (reutilização de código) . . . . .	74
2.11 Facilidade de Instalação (implantação) . . . . .	75
2.12 Facilidade Operacional . . . . .	75
2.13 Operação do Sistema em Múltiplos Locais . . . . .	76
2.14 Facilidade de Mudanças (Flexibilidade) . . . . .	76
3 Cálculo dos Pontos de Função Ajustados . . . . .	77
ANEXO B – Detalhamento da Métrica Pontos por Use Case (UCP) . . . . .	78
1 Total de pesos não ajustados dos atores . . . . .	78
2 Total de pesos não ajustados dos casos de uso . . . . .	78
3 Pontos totais não ajustados . . . . .	79
4 Determinar o fator de complexidade técnica . . . . .	79
5 Determinar a eficiência do fator ambiental . . . . .	80
6 Pontos totais de casos de uso ajustados . . . . .	81
7 REFERÊNCIAS BIBLIOGRÁFICAS . . . . .	82

APÊNDICE A – MODELO DE PESQUISA DE PROJETOS CONCLUÍDOS

APÊNDICE B – ATA DE REUNIÃO JAD

APÊNDICE C – DOCUMENTAÇÃO TÉCNICA DO SAGEP

## LISTA DE FIGURAS

FIGURA 2.1 – Representação de um Caso de Uso . . . . .	14
FIGURA 2.2 – Áreas de Conhecimento do PMBoK . . . . .	25
FIGURA 4.1 – Dinâmica de funcionalidade do SAGEP . . . . .	40
FIGURA 4.2 – Tela de cadastramento do escopo inicial do projeto . . . . .	41
FIGURA 4.3 – Tela de solicitação de mudança em um projeto . . . . .	42
FIGURA 4.4 – Tela de análise de uma solicitação de mudança . . . . .	42
FIGURA 4.5 – Tela de estimativa de impacto de um Use Case no projeto . . . . .	43
FIGURA 4.6 – Tela de avaliação de uma solicitação . . . . .	44
FIGURA 4.7 – Telas de registro e cálculo de Pontos de Função . . . . .	46
FIGURA 4.8 – Tela de consulta aos dados históricos de Use Cases . . . . .	47
FIGURA 4.9 – Tela de consulta detalhes de histórico . . . . .	48
FIGURA 4.10 – Tela de consulta aos recursos utilizados em um Use Case . . . . .	49
FIGURA 4.11 – Tela de avaliação do andamento do projeto . . . . .	49
FIGURA 4.12 – Tela de apuração de dados reais de um Use Case . . . . .	51
FIGURA 5.1 – Análise de resultados . . . . .	58
FIGURA A.1 – Funções de uma aplicação segundo a APF . . . . .	65

## LISTA DE TABELAS

TABELA 1.1 – Estrutura do Trabalho . . . . .	4
TABELA 3.1 – Funções do PRAXIS MENTOR . . . . .	31
TABELA 3.2 – Funções do SIGERE . . . . .	33
TABELA 5.1 – Erros de duração de atividades (previsto X real) . . . . .	55
TABELA 5.2 – Erros de duração de atividades (previsto X real) após o SAGEP .	57
TABELA A.1 – Complexidade das funções de dados (ALI / AIE) . . . . .	64
TABELA A.2 – Complexidade das entradas externas (EE) . . . . .	67
TABELA A.3 – Complexidade das saídas externas (SE) . . . . .	68
TABELA A.4 – Complexidade das consultas externas (CE) – ENTRADAS . . . .	69
TABELA A.5 – Complexidade das consultas externas (CE) – SAÍDAS . . . . .	69
TABELA A.6 – Fatores para cálculo de pontos de função brutos . . . . .	70
TABELA B.1 – Classificação dos atores . . . . .	78
TABELA B.2 – Classificação dos Casos de Uso . . . . .	78
TABELA B.3 – Avaliação dos fatores técnicos . . . . .	79
TABELA B.4 – Avaliação dos fatores ambientais . . . . .	80

## LISTA DE FÓRMULAS

FÓRMULA 2.1 – Pontos de Função Não Ajustados (NESMA) . . . . .	19
FÓRMULA A.1 – Valor do Fator de Ajuste da APF . . . . .	77
FÓRMULA B.1 – Total de Pontos Não Ajustados da UCP . . . . .	79
FÓRMULA B.2 – Fator de Complexidade Técnica da UCP . . . . .	80
FÓRMULA B.3 – Fator de Complexidade Ambiental . . . . .	80
FÓRMULA B.4 – Pontos totais de caso de uso ajustados da UCP . . . . .	81

## ABREVIATURAS E SIGLAS

Item	Descrição
ANSI	ANSI - American National Standards Institute
APF	Análise de Pontos de Função
BFPUG	Brazilian Function Point Users Group
CFPS	Certified Function Point Specialist
CMMI	Capability Maturity Model Integration
COSMIC	Common Software Measurement International Consortium
FFP	Full Function Points
GP	Gerente de Projeto
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IFPUG	International Function Point Users Group
ISO	International Organization for Standardization
JAD	Joint Application Design
PDP	Professional Development Program
PDU	Professional Development Units
PMBok	Project Management Body of Knowledge
PMI	Project Management Institute
PMP	Project Management Professional
SAGEP	Sistema de Apoio à Gestão do Escopo do Projeto
SEI	Software Engineering Institute
UCP	Use Case Points
UML	Unified Modeling Language

## **RESUMO**

Gerenciar projeto não é uma tarefa fácil de ser efetuada com sucesso por profissionais não especializados, principalmente pelas próprias características que definem um projeto como sendo um conjunto de atividades não rotineiras e com duração determinada. Gerenciar projetos de sistemas na área de Tecnologia da Informação, tem como complicação adicional o fato de seu produto ser intangível e muito sujeito a modificações do escopo. O presente trabalho tem como foco o controle dos requisitos que delimitam o escopo do projeto, pois um bom levantamento de requisitos, mesmo diminuindo a necessidade de mudanças, não impede que aconteçam, por diversas situações como mudanças de legislação, alteração de processos do negócio do cliente, etc. Para apoiar a gestão do escopo do projeto, foi desenvolvida uma ferramenta que auxilia no dimensionamento do impacto das solicitações de mudanças quanto à duração e o custo do projeto, baseado em dados históricos armazenados no desenvolvimento de projetos anteriores. Para possibilitar a comparação de tarefas com complexidades diferentes, é utilizada a métrica Pontos de Função. Quando ocorre uma solicitação de mudança, a decisão quanto à sua aprovação ou não é feita pelo cliente do projeto, que para tal, avalia os benefícios da mudança e os impactos previstos e no caso de aprovação, assume a alteração na duração e no orçamento do projeto. Após a construção, a ferramenta foi implantada de forma experimental e seus resultados foram coletados e analisados sendo apresentados neste trabalho.

**Palavras-chave:** Gerenciamento de Projetos, Engenharia de Requisitos, Métricas de software, Pontos de Função, PMI, Gerenciamento de Requisitos.

## **ABSTRACT**

To manage project is not an easy job of being effected successfully by professionals not specialized, mainly for the proper characteristics that define a project as being a set of not routine activities and with definitive duration. To manage projects of systems in the area of Technology of the Information, has as additional complication the fact of its intangible product to be and very subject to modifications of the scope. The present work has as focus the control of the requirements that delimit the scope of the project, because a good survey of requirements, exactly diminishing the necessity of changes, does not hinder that they happen for diverse situations as legislation changes, alteration of processes of the business of the customer and others. To support the management of the project, was developed a tool that assists in the sizing of the impact of the requests of changes as to the duration and the cost of the project, based in stored historical data in the development of previous projects. To make possible the comparison of jobs with different complexities is used the Function Point metrics. When a change request occurs, the decision as to its approval or not, it is made by the customer of the project, that to it evaluates the benefits of the change and impacts foreseen and in the case of approval, assumes the alteration in the duration and the budget of the project. After the construction, the tool was implanted of experimental form and its results had been collected and analyzed being presented in this work.



# 1 – INTRODUÇÃO

## 1.1 – Motivação

Gerenciar projetos corresponde à prática de definições, planejamento, controle e a conclusão dos mesmos. Conforme PMBoK (2004), um projeto é um empreendimento temporário, com o objetivo de criar um produto, serviço ou resultado único. Cada projeto tem início e fim bem definidos, com alocação provisória de recursos para resolver um problema diferente, com características que o distinguem de outros tipos de trabalho, podendo ser de vários tamanhos e tipos.

Conforme Jurison (1999), não obstante o tamanho, todos os projetos exibem características comuns que os distinguem de outros tipos de trabalho:

- Embora alguns projetos possam ter orçamentos folgados, devem ser terminados dentro de um orçamento dado;
- Os projetos são realizados por equipes com pessoas alocadas em tempo integral ou parcial, dependendo de suas necessidades específicas;
- Enquanto o grau de originalidade pode variar de projeto para projeto, todos são essencialmente diferentes e não rotineiros.

Quando se trata de projetos de desenvolvimentos de softwares, o quadro também mostra-se preocupante há algumas décadas. Segundo Wilson (1998), uma estatística realizada em 1989 nos Estados Unidos mostrou que 29% dos projetos de software iniciados nunca foram entregues e 47% apesar de concluídos nunca foram utilizados. Entre os demais, apenas 5% sofreram pouca ou nenhuma alteração após a implantação.

Segundo Chaos (2003), o Standish Group, fez uma comparação entre os resultados de pesquisas realizadas de 1994 a 2003, mostrando que a taxa de projetos de Tecnologia da Informação (T.I.) bem sucedidos aumentou de 16% no início desse período para 34%, entre os 13.522 projetos pesquisados no último ano. A taxa de projetos abortados caiu de 31% para 15% no mesmo período. Em termos financeiros essa mesma pesquisa revelou que em 2003 a estimativa do valor desperdiçado em T.I. foi da ordem de US\$ 55 bilhões em um total de US\$ 250 bilhões gastos. Com relação ao cumprimento de compromissos, a duração dos projetos em média supera

em 82% o tempo estimado e em 52% dos projetos, alguns requisitos solicitados na definição do escopo, ficam “para versões futuras”.

Esses números, embora ainda bastante assustadores, tiveram uma relativa melhora devido à crescente preocupação nos últimos anos com a padronização, criação de normas e estudo de melhores práticas em gerenciamento de projetos.

Conforme relatado em Fernandes (1992), alguns dos motivos que levam os projetos de T.I. ao insucesso são:

- Dificuldade em captar os requisitos dos usuários gerando especificações incompletas;
- Dificuldade em estimar prazos e recursos, adotando-se normalmente o método da “chutologia”, sendo que poucas empresas utilizam estimativas baseadas em modelos matemáticos e registros históricos;
- Falta de marcos concretos ao longo do projeto, a fim de verificar seu avanço;
- Metodologias inadequadas ou inexistentes.

Em Pressman (2006) após tentativas de identificação de possíveis fatores que geram as principais dificuldades no desenvolvimento de software, fica evidente a convergência das causas para o despreparo da maioria dos gerentes de projetos de sistemas, com perguntas que parecem ser a chave para a solução de alguns dos principais problemas:

- Por que existe tanta dificuldade em medir o progresso do software durante seu desenvolvimento?
- Por que demora tanto tempo para que os sistemas sejam concluídos?
- Por que todos os erros não são descobertos e sanados antes da entrega do sistema para o usuário?
- Por que os custos são tão elevados?

Esse ponto de vista é compartilhado por diversos especialistas em Gerenciamento de Projetos, sendo inclusive citado em Jurison (1999) com a afirmação de que algumas das dificuldades existentes são de natureza inerente ao produto, outras são relacionadas ao gerenciamento. Entre os problemas comuns relacionados ao software estão:

- Intangibilidade - O software é intangível. É difícil de controlar porque não contém marcos visíveis para medir o progresso e a qualidade.

- Complexidade - A complexidade do software dificulta sua compreensão, criando não somente problemas técnicos, mas também de gerenciamento.
- Volatilidade dos Requisitos - Os requisitos do software são constantemente mudados.

## **1.2 – Objetivos**

Este trabalho tem como objetivos:

- Propor uma ferramenta para auxiliar a gestão de solicitações de mudanças de requisitos que geralmente ocorrem no transcorrer do desenvolvimento, dando suporte às estimativas de prazo e custo de tais mudanças, através da estratificação de dados históricos de funções similares;
- Medir os benefícios dessa ferramenta em um ambiente real de desenvolvimento de sistemas, implementando a ferramenta proposta no gerenciamento de alguns projetos e comparando os resultados obtidos com dados coletados em outros projetos encerrados sem a utilização dessa ferramenta.

O foco da ferramenta proposta foi obtido a partir da experiência vivenciada na área de desenvolvimento de sistemas de uma empresa escolhida como estudo de caso, onde foram coletados dados sobre os projetos desenvolvidos e identificadas as principais dificuldades no gerenciamento dos projetos que não obtinham o nível de sucesso esperado.

Conforme o Guia do Conjunto de Conhecimentos em Gerenciamento de Projetos editado pelo PMI - Project Management Institute, PMBoK (2004), o conhecimento dos dados históricos é fundamental para a gestão das diversas áreas de conhecimento. Daí surgiu o ponto de partida para a criação de uma base histórica com os dados levantados dos projetos da empresa utilizada no estudo de caso, transformando-os em informação de suporte para apoio ao Gerente de Projetos. Essa informação, aliada ao conhecimento das técnicas de gerenciamento, transformar-se-á no diferencial para a esperada melhoria dos resultados.

### 1.3 Estrutura do trabalho

A Tabela 1.1 apresenta a estrutura dos capítulos deste trabalho, no qual, a partir da contextualização do problema apresentado neste primeiro capítulo, os dois seguintes resumem uma revisão bibliográfica efetuada sobre o assunto.

O Capítulo 2 apresenta algumas técnicas atualmente utilizadas, visando minimizar as falhas de projeto de softwares, tanto na arquitetura de sistemas, quanto nas estimativas necessárias para o correspondente planejamento, enquanto no Capítulo 3 são apresentados alguns trabalhos de pesquisa relacionados ao estudo em questão.

O Capítulo 4 apresenta a arquitetura da ferramenta de apoio desenvolvida para auxiliar o gerenciamento do escopo do projeto e as estimativas de duração e custos de um projeto, mostrando as principais funcionalidades do sistema.

O Capítulo 5 apresenta o estudo de caso utilizado e um resumo do processo de desenvolvimento e implantação em alguns projetos dessa empresa da ferramenta desenvolvida.

O Capítulo 6 apresenta uma análise dos resultados obtidos nos projetos que utilizaram essa ferramenta na empresa em questão, mostrando os benefícios alcançados e comparando tais resultados com os de projetos que foram desenvolvidos da forma anterior, na mesma empresa.

O Capítulo 7 apresenta a Conclusão deste trabalho, apresentando sugestão de trabalhos futuros relacionados à pesquisa apresentada.

CAPÍTULO	TÍTULO
1	Introdução
2	Fundamentos
3	Trabalhos Relacionados
4	Arquitetura: Sistema de Apoio à Gestão do Escopo do Projeto - SAGEP
5	Estudo de Caso
6	Conclusão
7	Referências Bibliográficas

TABELA 1.1 – ESTRUTURA DO TRABALHO

## **2 – FUNDAMENTOS**

### **2.1 – Engenharia de Requisitos**

Conforme o IEEE Standard Glossary of Software Engineering Terminology (1983), um requisito é uma condição ou capacitação necessária para solucionar um problema. É uma condição ou capacitação que um sistema ou componente do sistema precisa atender ou ter para satisfazer um contrato, padrão, especificação, ou outro documento formalmente estabelecido. O conjunto de todos os requisitos delimita o escopo do projeto e forma a base para o posterior desenvolvimento do sistema.

A perfeita identificação dos requisitos é o ponto de partida para se atingir o sucesso de um sistema. Conforme Kotonya (1998), em desenvolvimento de software, essa tarefa não é nada fácil, não se podendo assumir a situação ideal de receber do usuário todas as informações necessárias para a construção do sistema, isto porque normalmente essas respostas nem mesmo ele conhece. Assim, cresce a importância do processo de descoberta dos requisitos verdadeiros.

#### **2.1.1 – Levantamento de Requisitos**

Corresponde a identificar junto aos clientes, usuários e outros envolvidos, quais são os objetivos do sistema ou produto. Segundo Kotonya (1998), apesar de parecer simples, trata-se de um processo extremamente complexo. Algumas das razões dessa dificuldade relacionam-se com problemas de escopo, problemas de compreensão e problemas de volatilidade.

Fiorini (1998) define como sendo o objetivo principal dessa fase “*concluir com êxito um acordo entre quem solicita e quem desenvolve, estabelecendo clara e rigorosamente o que deverá ser produzido*”.

Existem diversas técnicas de levantamento de requisitos, tais como Entrevistas, Questionários, Observação Direta, Pesquisa Bibliográfica, JAD – Joint Application Design, etc., que entre si, não são incompatíveis, podendo ser utilizadas até mesmo simultaneamente.

### **2.1.1.1 Entrevistas**

Tem como objetivo a coleta de informações sobre o comportamento de um sistema atual ou sobre os requisitos de um novo sistema, de pessoas que conheçam o processo pesquisado. Serve para o analista verificar sua compreensão sobre o assunto. Segundo Chinelato (2004), deve consistir em um diálogo entre as partes, seguindo um roteiro previamente elaborado pelo entrevistador, adaptado ao perfil do entrevistado. Visa diminuir a resistência natural das pessoas a mudanças, tornando-as co-responsáveis pelas decisões e portanto mais dispostas a aceitá-las e defendê-las.

A entrevista deve ser agendada com antecedência, após aval do chefe do entrevistado. Yourdon (1990) sugere como estrutura básica de uma entrevista:

- Preparar a si e ao entrevistado antes e no início da entrevista;
- Apresentar-se, partir do geral para o específico e manter um ritmo adequado;
- Achar o porquê das coisas usando: O Que, Quando, Como;
- Controlar a duração e terminar em acordo de ação posterior;
- Agradecer a colaboração e colocar-se à disposição.

Yourdon (1990) alerta também para problemas comuns em entrevistas, para os quais deve-se estar preparado:

- Entrevistar a pessoa errada na hora errada - Contrapartes do setor, nem sempre são as pessoas com mais conhecimentos sobre o assunto. Mesmo que encontre a pessoa certa, pode-se estar entrevistando-o durante um período em que o usuário está mergulhado sob outras pressões e emergências.
- Fazer perguntas erradas e receber respostas erradas - Mesmo usando um único idioma, é comum o analista e os usuários terem vocabulários diferentes, experiências básicas, percepções, valores e prioridades distintas. Com isso é fácil se fazer uma pergunta e obter uma resposta sem que o interlocutor tenha entendido a pergunta, ou o analista achar que entendeu a resposta de forma diferente da emitida, sem que ambos percebam o engano.
- Criar ressentimentos recíprocos - O usuário pode interpretar a entrevista como o primeiro passo para que venha a perder seu emprego e criar uma barreira de comunicação. O analista pode não entender plenamente do assunto e achar que há pré-disposição do usuário em lhe omitir informações.

### 2.1.1.2 Questionários

Uma grande vantagem de se utilizar questões livres de contexto é que as mesmas podem ser preparadas com antecedência além de ajudarem a vencer o embaraço inicial relacionado com um novo projeto e novas relações.

Segundo Cordeiro (2002), algumas reflexões devem ser feitas antes do início dos trabalhos, independentemente do produto a ser desenvolvido e da utilização de outra técnica complementar de levantamento de requisitos. A partir do conhecimento das respostas, deve ser traçado o Plano do Projeto. Algumas destas questões são:

- Quem é o cliente do projeto ?
- Qual o problema que o cliente deseja resolver ?
- Qual é a razão real para desejar solucionar o problema ?
- Quem deve estar na equipe do projeto ?
- Quanto tempo teremos para desenvolver este projeto ?
- Existe mais alguém que possa dar informações úteis ?
- Existe algum material adicional que possa conter informações úteis ?

As perguntas podem ser divididas em duas categorias:

- Perguntas fechadas - Em geral, exigem respostas monossilábicas como “Sim” e “Não” e o trabalho maior é da pessoa que pergunta. Seu objetivo principal é obter itens específicos de informação. Perguntas iniciadas por “Quem, Onde, Quando e Quantos” também são perguntas fechadas.
- Perguntas Abertas - Estimulam o entrevistado a pensar, falar e expressar suas idéias com suas próprias palavras. Um perigo em perguntas abertas é o desvio do assunto objetivado, devendo o entrevistador estar atento para evitar que isso aconteça e estar preparado para saber interpretar as respostas. Uma grande vantagem é que freqüentemente nas respostas abertas, são captadas informações que vão muito além do que foi formulado na pergunta.

Segundo Chinelato (2004), o questionário é mais formal, concreto e frio que as entrevistas, podendo gerar distorções e falseamento da realidade. É mais recomendável quando as pessoas estão dispersas em locais afastados, quando as perguntas forem bastante objetivas ou necessitarem de pesquisas para serem confiáveis ou quando não houver tempo suficiente que viabilize a entrevista pessoal.

### **2.1.1.3 Observação Direta**

Quando vemos algo, normalmente enriquecemos e validamos as conclusões tiradas através das outras técnicas de levantamentos. Segundo Gomoll (1990) a observação direta, por não ser um experimento, não tem como objetivo traduzir ações do usuário em dados estatísticos, e sim coletar dados úteis ao desenvolvimento do sistema e pode ser descrita através dos seguintes passos básicos:

- **Preparação** - A observação requer uma preparação onde será estabelecido o conjunto de tarefas mais críticas do sistema que deverão ser observadas;
- **Apresentação** - Os usuários devem conhecer o papel do analista e o processo do qual eles estão participando. A ênfase é não intimidar o usuário deixando claro que o alvo é o entendimento do processo e não o desempenho do usuário;
- **Ruptura** - Outro ponto importante é deixar o usuário à vontade para encerrar sua participação quando for conveniente;
- **Demonstração** - Se for utilizar algum equipamento extra para registro da observação, como câmeras fotográficas, gravadores ou filmadoras é bom avisar com antecedência para não deixá-lo intimidado durante a observação;
- **Aplicar a técnica de “pensar em voz alta”** - Desta forma, os usuários devem ser instruídos a falar aquilo que pensam durante a execução das tarefas;
- **Perguntas** - Verificar se os usuários ainda apresentam alguma dúvida sobre o processo de observação ou sobre o seu papel, antes de iniciar o processo;
- **Conclusão** – Ao final deve-se agradecer a participação dos usuários e questioná-los se algum processo que consideram importante não foi observado;
- **Análise** - Deve ser detalhada e centrar-se na identificação dos problemas e então, verificar as causas e possíveis soluções do mesmo.

### **2.1.1.4 Pesquisa Bibliográfica**

As informações coletadas nessa pesquisa são provenientes de livros técnicos, estudos e documentos observados. Conforme Chinelato (2004) se for realizada antes dos outros meios de levantamento, proporciona melhor orientação nas entrevistas, perguntas do questionário melhor elaboradas e compreensão mais coerente nas observações realizadas. Essa técnica também é importante após outros tipos de levantamentos, para aprofundar e consolidar o entendimento adquirido.



### 2.1.1.5 – JAD - Joint Application Design

Segundo August (1993) e Costa (1994), JAD é uma técnica de levantamento de requisitos desenvolvida pela IBM em 1977 no Canadá e atualmente utilizada em todo o mundo por empresas que visam acelerar o desenvolvimento de sistemas para computador, com mais eficácia e em menor tempo. Tem como objetivo extrair informações dos especialistas no negócio, através de reuniões ou sessões de trabalho, utilizando a criatividade, técnicas visuais e dinâmica de grupo, através de reuniões estruturadas e objetivas que buscam decisões em consenso.

A técnica JAD define personagens que devem ser indicados logo na abertura do projeto:

- Executivo Patrocinador. Também chamado *Sponsor* do projeto, deve ter posição gerencial dentro da empresa cliente, superior aos demais integrantes do projeto, com poder de decisão e com influência nos altos escalões da empresa. Estabelece os objetivos e diretrizes do projeto, sendo responsável pela abertura da primeira reunião. Ao final deverá avaliar e aprovar o resultado dos trabalhos;
- Gerente do Projeto. Deve ser declarado logo na fase inicial do projeto, conforme PMBoK (2004), devendo estar plenamente capacitado para conduzir o projeto em questão. Deverá também viabilizar a realização dos projetos complementares, nos prazos estipulados;
- A Equipe do projeto. É formada por Analistas de Sistemas e pelo menos um usuário de cada área envolvida com o projeto;
- Líder de Sessão. Também chamado de Facilitador, deve dominar as técnicas de dinâmica de grupo, gestão de conflitos e relações pessoais, funcionando como um maestro da equipe, estimulando e inspirando o trabalho em harmonia;
- Documentador. Deve ter capacidade de síntese, facilidade na redação de atas, sendo o responsável em documentar as decisões tomadas nas reuniões, indicando os principais tópicos discutidos que levaram o grupo ao consenso.
- Observadores. Podem fazer parte da sessão JAD, não se obrigando a estarem presentes em todas as reuniões, não participando ativamente das mesmas. Podem ser estagiários ou funcionários novos que participam a fim de conhecerem os detalhes dos setores envolvidos ou apenas observar a técnica.

Durante as reuniões, o Líder do JAD deve ter habilidade suficiente para coordenar as discussões evitando atritos pessoais, balancear a participação dos integrantes, cortar conversas paralelas, envolvendo todos os integrantes no assunto e controlar o cumprimento dos horários estabelecidos no início da sessão.

Cada tópico discutido deve ser fechado através da definição em consenso antes de se passar para outro tópico. O Líder registra no *flipchart* a decisão obtida à frente de todos, e ao completar cada folha, ao invés de virá-las para trás como normalmente acontece, estas devem ser destacadas e presas com fita crepe nas paredes em toda a volta do local da reunião, evitando assim que cada integrante anote suas conclusões e permitindo uma clara visualização de tudo o que foi decidido, até a última reunião, para evitar retomadas de discussões em assuntos já consensados.

### **2.1.2 – Análise e Negociação de Requisitos**

Os requisitos levantados formam a base para a análise de requisitos. No Levantamento de Requisitos o foco foi dado pelos usuários e clientes. Já na fase de Análise e Negociação será destacada a visão dos desenvolvedores, embora ainda não se focalize as soluções e sim a definição dos problemas a serem solucionados.

Esta análise categoriza e organiza os requisitos em subconjuntos relacionados, explora o relacionamento de cada requisito com todos os demais, examina consistência, omissão e ambigüidade dos requisitos e os prioriza com base nas necessidades dos usuários.

Segundo Kotonía (1998), deve-se verificar se todos os problemas levantados pelos *stakeholders* e consensados como pertinentes ao escopo do projeto foram contemplados na relação dos requisitos e se ocorrem conflitos ou sobreposição entre os mesmos, gerando-se um documento de requisitos consensados.

A utilização de protótipos pode facilitar muito a negociação de requisitos, pois o usuário final poderá avaliar mais concretamente as funcionalidades propostas, antes do esforço da construção das funções do sistema. Sua elaboração não deve ser demorada e focar apenas a funcionalidade desejada, podendo posteriormente ser aproveitada na versão inicial do sistema. A facilidade de se fazer ajustes e alterações sugeridas durante a negociação é evidente, visto que não foi construído ainda o sistema que suportará as interfaces apresentadas, seus modelos de dados e a interação

com outros sistemas. Tudo que é mostrado é apenas fachada, podendo até mesmo haver simulações de funcionalidade, desde que sejam de baixa complexidade.

O trabalho dessa etapa pode ser bastante facilitado por um *Checklist*, como sugere Sommerville (1997):

- Os requisitos estão alinhados com as metas do cliente?
- Os requisitos são realmente necessários ao projeto ou são apenas desejos de quem os definiu?
- A descrição efetuada indica o agrupamento de mais que um requisito que pode ser desdobrado?
- O requisito pode ser interpretado de forma ambígua por outro *stakeholder*?
- Os requisitos podem ser devidamente testados?
- É possível a implementação do requisito no sistema?

### **2.1.3 – Especificação de Requisitos**

Uma especificação pode ser um documento escrito, um modelo gráfico, um modelo matemático formal, uma coleção de cenários de uso, ou qualquer combinação dos itens citados. Para grandes sistemas, um documento escrito contendo linguagem natural combinada a modelos gráficos pode ser a melhor abordagem. Para sistemas pequenos desenvolvidos em ambiente técnico conhecido pode ser suficiente a utilização de cenários de uso.

Pode ser criado um modelo do sistema, contendo as interfaces com o usuário, as entradas, saídas, funções e controles do sistema, e qualquer outro artefato que possa auxiliar na especificação. Pode-se também definir a partir de um diagrama de contexto, as entradas e respectivos fornecedores de informação e as saídas com os correspondentes destinatários que interagem com o sistema.

O resultado, devidamente documentado, servirá de base para a especificação das necessidades de hardware, software e banco de dados, devendo portanto estar padronizado de acordo com o entendimento dos demais profissionais que utilizarão esse documento.

#### **2.1.4 – Validação de Requisitos**

A validação examina a especificação para garantir que todos os requisitos do sistema foram estruturados de maneira não ambígua, que as inconsistências, omissões e erros foram identificados e corrigidos, e que os produtos de trabalho estão em conformidade com os padrões estabelecidos para o processo, para o projeto e para o produto.

Kotonya (1998) recomenda que essa validação seja efetuada por usuários, clientes e outros analistas de sistemas, preferencialmente não sendo os responsáveis pela especificação dos mesmos. Normalmente deve ser formalizada e documentada através de uma Revisão Técnica, observando a incidência de informações inconsistentes, ambigüidades, requisitos conflitantes ou requisitos irreais. Verifica também se são devidamente rastreáveis, se a fonte está identificada, se viola alguma regra de domínio e se é passível de ser testado e implementado.

#### **2.1.5 – Gerenciamento de Requisitos**

Os requisitos mudam devido às alterações que ocorrem no ambiente do sistema e conforme os clientes desenvolvem um melhor entendimento de suas reais necessidades. Novos requisitos surgem e há alterações nos requisitos em todos os estágios do processo de desenvolvimento do sistema. Em muitos casos não se pode evitar tais mudanças. Segundo Kotonya (1998) são comuns os casos em que muitos requisitos são alterados antes que o sistema seja posto em operação, o que causa sérios problemas para os desenvolvedores.

Os requisitos devem ser documentados e controlados, minimizando assim as dificuldades decorrentes das mudanças. Fiorini (1998) destaca a importância da criação de uma *baseline* de requisitos, isto é, um conjunto de artefatos aceitos e gerenciados, que servirão de linha de ação no projeto. Após seu estabelecimento, qualquer alteração só pode ser efetuada após um procedimento formal, o que permite um maior controle sobre as alterações e versões dos requisitos que formam o escopo do projeto. Quando isso não ocorre, é comum a incidência de mudanças que não são essenciais, muitas das quais depois de aceitas e implementadas, mostram-se incoerentes e são desfeitas, gerando um atraso no prazo e aumento do custo.

Segundo Kotonya (1998), uma pesquisa realizada na Europa, em mais de 4000 empresas, mostrou que a maioria dos problemas ocorridos no desenvolvimento e produção de software, tem sua causa raiz no gerenciamento dos requisitos.

Um requisito só pode ser gerenciado de forma efetiva se for rastreável, isto é, se for possível identificar quem solicitou o requisito, o que o fez existir, qual o seu relacionamento com outros requisitos e como se relaciona com outras informações ou documentos existentes. Isso é fundamental para que seja possível identificar o impacto de uma solicitação de mudança no projeto.

Kotonya (1998) alerta também sobre o perigo de se menosprezar o gerenciamento de requisitos, por parecer uma despesa desnecessária pois não dá retornos imediatos, porém tal economia de tempo e custo pode ser a causadora de um grande retrabalho, gerando um grande aumento de custo e prazo no decorrer das fases seguintes do projeto, além da qualidade que poderá ser comprometida pela falta de um total controle do escopo.

## **2.2 – USE CASE (Caso de Uso)**

Conforme Jacobson (1992), um caso de uso é um "*documento narrativo que descreve a seqüência de eventos de um ator que usa um sistema para completar um processo*".

Caso de Uso é uma técnica de modelagem que demonstra o que um sistema deve fazer. Não descrevem como o software deverá ser construído, e sim, como ele deverá se comportar. É constituído de um processo iterativo, através de discussões buscando o entendimento entre o cliente e os desenvolvedores, conduzindo a uma especificação do sistema. Tem como objetivos:

- Decidir e descrever os requisitos funcionais do sistema;
- Descrever clara e consistentemente o que o sistema deve fazer;

Os casos de uso foram propostos inicialmente por Jacobson (1992) em sua metodologia de desenvolvimento de sistemas orientados a objetos. Pode ser representado por uma elipse contendo o nome do caso de uso no seu interior. Os elementos gráficos que representam atores e casos de uso são mostrados na Figura 2.1 [Larman (2004)].

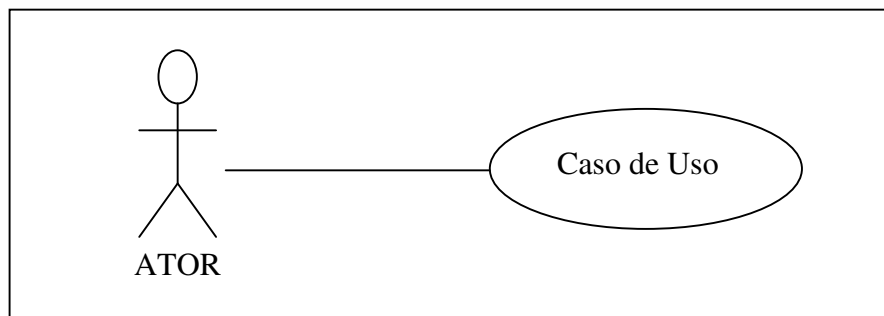


FIGURA 2.1 – REPRESENTAÇÃO DE UM CASO DE USO

### 2.3 – Diagrama de Transição de Estado (DTE)

A idéia básica do Diagrama de Transição de Estado, é estudar certos tipos de lógicas que envolvem transições possíveis entre diferentes estados. Segundo Page-Jones (2001) é uma ferramenta ideal para atributos que possuem poucos valores e têm restrições em transições autorizadas entre esses valores.

Para a elaboração de um DTE, algumas definições são necessárias:

- Um ESTADO representa uma situação, um cenário ou um modo de comportamento de um sistema ao ser observado em determinado momento. É representado por um retângulo com a identificação do estado.
- Uma TRANSIÇÃO representa a passagem do sistema de um estado para outro. É representada por uma seta ligando um estado a outro.
- Uma CONDIÇÃO representa a causa necessária para que haja a transição de estado. Decorre da ocorrência de um evento ou circunstância que propicia a transição de estado. Escrita ao lado da ação, acima de uma linha horizontal.
- Uma AÇÃO representa a atividade do sistema que efetua a transição de estado. Pode ser escrita abaixo da linha onde está definida a ação.
- O ESTADO INICIAL é o estado atribuído a um objeto quando este é criado. É representado por um círculo preenchido.
- O ESTADO FINAL indica o fim do ciclo de vida, podendo existir mais que um para um mesmo objeto. É representado por um “olho de boi”.

Um objeto muda de estado quando acontece algo, o fato do objeto alterar o seu estado chamamos de evento. Analisando as mudanças de estado que um objeto

pode sofrer, podemos prever todos os possíveis comportamentos de um objeto de acordo com os eventos que o mesmo possa sofrer.

## 2.4 – Métricas de Software

Um dos instrumentos essenciais para o exercício das funções do Gerente de Projetos na área de Tecnologia da Informação, é o conhecimento de dados quantitativos sobre os sistemas em desenvolvimento e seus próprios processos. Esse conhecimento é adquirido através de técnicas denominadas "métricas de software" que estão dispersas nas últimas décadas em diversas publicações, tais como Fernandes (1995) e Dekkers (2003).

Para estimar esforço e prazo, é preciso que seja selecionada uma abordagem para a obtenção de estimativas. Conforme McGarry (2002), as abordagens existentes podem ser divididas em:

- Modelos Paramétricos – Assumem a existência de uma relação matemática entre tamanho, esforço e prazo afetada por parâmetros de desempenho. Os relacionamentos são baseados em suposições teóricas e/ou dados históricos armazenados de experiências anteriores.
- Modelos Baseados em Atividades – Também chamada estimativa *bottom-up*, esta modalidade consiste em enumerar todas as atividades do projeto e estimar o esforço e prazo para cada uma delas.
- Analogia – Esta técnica baseia-se na comparação das características do projeto com a de outros projetos concluídos. As diferenças são identificadas, sendo introduzidas as mudanças necessárias para produzir as estimativas. Algumas das categorias desta técnica são:
  - Adequado para projetos pouco freqüentes onde não existirá histórico;
  - Um ou mais projetos semelhantes servirão de base para a estimativa;
  - Identificar detalhadamente as diferenças;
  - Podem ser utilizadas as mesmas técnicas de estimativa dos projetos-modelo, ajustadas para as diferenças encontradas.
- Relações Simples de Estimativas – Trata-se de uma simplificação dos modelos paramétricos. Utilizam-se relações matemáticas simples, baseadas em dados históricos locais, ao invés de modelos matemáticos abrangentes. De

uma forma geral, os relacionamentos deste tipo não são aplicáveis a organizações e contextos diferentes dos utilizados para a coleta dos dados.

- Devem ser usadas para simplificar o processo e não por desconhecimento de outras opções;
- Exigem uma base histórica considerável e um processo estabilizado de desenvolvimento;
- Devem ser utilizadas dentro das suposições usadas na calibragem;

De acordo com Kemerer (1987), para se utilizar qualquer método de estimativa é necessário coletar dados históricos dos projetos da instalação, a fim de que os modelos sejam calibrados às condições locais.

Dentre dezenas de técnicas existente, são apresentadas a seguir algumas das mais conhecidas.

#### **2.4.1 – LOC – *Lines Of Code* (Linhas de Código Fonte)**

As primeiras métricas de estimativa de tamanho de software surgiram entre 1950 e 1960 e se basearam no tamanho físico de linhas de código como o *Lines of Code*, sendo orientada ao tamanho do programa fonte. Embora esta métrica possa parecer simples, existe discordância sobre o que constitui uma linha de código. Conforme Roetzheim (1998), a medida de linhas de código não deveria contar linhas de comentário e linhas em branco, uma vez que estas servem para a documentação interna do programa e não afeta a sua funcionalidade. A regra de cálculo mais universal é a de que todas as linhas que não sejam, nem de comentário, nem estejam em branco, sejam consideradas, independentemente do número de instruções por linha.

Segundo Dias (2003) a métrica LOC apresenta dificuldades para uso em comparações de sistemas desenvolvidos por linguagens diferentes, apresenta distorções para efeito de comparação, por ser dependente do estilo dos programadores além de apresentar impossibilidade de uso em estimativas, considerando que os sistemas somente poderiam ser medidos depois de codificados.



### **2.4.2 – APF – Análise de Pontos de Função**

Segundo Dekkers (2003) o conceito da técnica de medição por pontos de função, foi introduzido por Allan Albrecht da IBM em 1979 sendo posteriormente refinado em uma metodologia formal e publicado no domínio público em 1984. Dois anos depois foi formado o Grupo Internacional de Usuários de Pontos de Função (IFPUG), como um grupo formalmente constituído e sem fins lucrativos visando efetuar padronizações adicionais nas regras da APF.

Pontos de Função são independentes dos métodos físicos, ferramentas ou linguagem de desenvolvimento utilizada para construir o software, podendo ser comparado na construção civil com a medição da área construída de uma casa. Assim como a quantidade de metros quadrados não corresponde à velocidade da construção da obra, o tamanho em Pontos de Função não mede a produtividade ou o esforço de seu desenvolvimento. Conforme Dekkers (2003), Pontos de Função medem o tamanho do que o software faz, independentemente de como ele é desenvolvido e implementado.

Segundo Hazan (2001), Pontos de Função é uma medida de dimensionamento de software através da funcionalidade implementada em um sistema, sob o ponto de vista do usuário, tendo como algumas de suas vantagens:

- relatar diretamente os requisitos do negócio;
- ser independente da tecnologia utilizada;
- possibilitar as estimativas nas fases iniciais do desenvolvimento de software;
- facilitar uma reestimativa;
- apoiar a análise de produtividade e qualidade;
- fornecer suporte ao gerenciamento do projeto.

Dias (2003) destaca que através desta técnica é possível calcular o esforço despendido por unidade, ou por atividade no departamento de TI e ainda, fornecer subsídios para melhor compreensão das correções, falhas e dos problemas de planejamento dos projetos já concluídos ou em andamento, de forma a desenvolver projetos futuros de maneira mais eficiente.

Em 2002 essa técnica tornou-se padrão internacional, através da norma ISO/IEC 20926, podendo ser conhecida mais detalhadamente em ISO (2003), porém na norma não é realizado o Cálculo do Fator de Ajuste que na APF ajustam o total de

Pontos de Função Não Ajustados, através de 14 Características Gerais de Sistemas. Segundo Calazans (2005) o ajuste proposto pela APF é atualmente questionado devido à grande variação na interpretação e a constatação que algumas delas estão desatualizadas para a maioria das plataformas atuais.

O detalhamento da métrica Pontos por Função e das 14 Características Gerais de Sistemas que ajustam os Pontos de Função são apresentados no Anexo A.

#### **2.4.3 – COSMIC-FFP**

Em 1999 um grupo de pesquisadores da Universidade de Quebec desenvolveu um novo método de medição funcional para sistemas de tempo real, denominado FFP (*Full Function Points*).

Em 1998 um grupo de especialistas em medição de software constituiu o COSMIC (*Common Software Measurement International Consortium*) com o objetivo de desenvolver um novo método de medição de tamanho funcional baseado nas melhores características dos métodos existentes e que incorporasse novas idéias.

Conforme Abran (2003), o método proposto em 2000 denominado COSMIC-FFP, na prática foi um refinamento do método FFP oficializado em 2003 através da ISO/IEC 19761:2003.

#### **2.4.4 – NESMA - Nederlandse Software Metrieken Associatie**

Um método de estimativa baseado em Pontos de Função bastante utilizado é a Contagem Indicativa da NESMA, associação holandesa fundada em 1989, sendo um dos primeiros grupos de usuário de APF no mundo. Mantém seu próprio manual de contagem, atualmente na versão 2.1, cuja primeira versão em 1990 foi baseada no manual do IFPUG, conforme Nesma (2005). Utiliza a mesma filosofia, conceitos, termos e regras do IFPUG, com algumas diretrizes diferentes. A estimativa de tamanho de uma aplicação, em Pontos de Função Não Ajustados (PFNA) é recomendada a partir da equação apresentada na Fórmula 2.1

$$PFNA = (35 * I) + (15 * E)$$

onde : PFNA = Pontos de Função Não Ajustados;

I = Grupos de dados Internos;

E = Grupos de dados Externos.

FORMULA 2.1 – PONTOS DE FUNÇÃO NÃO AJUSTADOS (NESMA)

Neste método, são identificados os grupos de dados relativos à natureza do negócio, segundo a perspectiva do usuário. Tais grupos são classificados como: Internos (I), se mantidos pela aplicação objeto do estudo; ou Externos (E), se apenas referenciados ou consultados pela aplicação.

Conforme Aguiar (2002), os fatores 35 e 15 da equação acima foram obtidos a partir de considerações sobre a relação entre os diferentes componentes do modelo da APF não sendo utilizados procedimentos estatísticos.

#### 2.4.5 – UCP – Use Case Points (Pontos por Caso de Uso)

Segundo Ribu (2001) os Pontos por Caso de Uso foram criados por Gustav Karner em 1993, como uma adaptação específica dos Pontos de Função, definindo como base de produtividade associada ao desempenho de 1 UCP, o esforço de 20 homens hora. A métrica UCP tem sido estudada nos últimos anos por vários pesquisadores no meio acadêmico e na indústria, embora ainda seja pouco divulgada.

Segundo Aguiar (2003) as contagens de UCP podem variar entre organizações e indivíduos, devido à variação nos estilos de casos de uso, fazendo com que as estimativas de produtividade também variem bastante. Dessa forma, a obtenção de previsões confiáveis de esforço exigiria a padronização dos estilos de casos de uso e um extenso trabalho de calibração do modelo de estimativas.

A inexistência de tais padrões universais dificulta a comparação entre projetos de diferentes organizações. Não há como garantir que os UCP estarão medindo a mesma coisa se os critérios utilizados forem muito diversificados.

O detalhamento da métrica Pontos por Use Case é apresentado no Anexo B.

#### **2.4.6 – COCOMO - Cost Constructive Model**

COCOMO é um modelo que permite estimar o custo, esforço e prazo durante o planejamento do desenvolvimento de um software, de acordo com práticas que eram comumente utilizadas nas décadas de 70 e 80. Foi desenvolvido pela University of Southern California em 1981, como resultado de estudos do Prof. Dr. Barry Boehm sobre 63 projetos de grande porte. Atualmente denominado COCOMO 81, parte do tamanho do produto expresso em linhas de código-fonte. A previsão do esforço e do prazo aumenta proporcionalmente ao tamanho.

Conforme Boehm (1981) existem três formas do modelo, cada uma oferecendo maior detalhe e acurácia, ao se avançar no processo de planejamento:

- Modelo Básico - Para equipes relativamente pequenas, desenvolvendo software em um ambiente doméstico, bastante conhecido;
- Modelo Intermediário - Quando a equipe é composta por técnicos experientes com relação a apenas alguns aspectos do sistema a ser desenvolvido;
- Modelo Detalhado - Quando o projeto precisa operar em um ambiente complexo de hardware, software, regras e procedimentos operacionais, como acontece nos sistemas em tempo real.

Há quatro categorias de direcionadores de custo que são consideradas significativas na previsão do desempenho de um projeto de desenvolvimento de software. Cada categoria possui diversos direcionadores de custo. A cada direcionador deve ser atribuído um grau: Muito Baixo, Baixo, Normal, Alto ou Extra Alto, que são utilizados no modelo a fim de ajustar, para cima ou para baixo, a estimativa do esforço básico de desenvolvimento.

As categorias de custo definidos no COCOMO são:

- Atributos do Produto;
- Atributos Computacionais;
- Atributos do Pessoal;
- Atributos do Projeto;

Deve ser tomado cuidado especial na utilização de COCOMO 81, uma vez que as premissas do modelo podem não se aplicar à situação específica do usuário, como por exemplo, o desenvolvimento incremental do projeto.

#### **2.4.7 – COCOMO II**

Em 1994 foram iniciadas as pesquisas para este novo modelo na University of Southern Califórnia-USC. Conforme Boehm (2000), buscou-se não apenas definir um novo modelo, mas também produzir o ferramental necessário para a aplicação da métrica. Em 1997 conseguiram implementar o software intitulado “USC COCOMO II”, cuja versão era denominada “1997.0”, a qual foi obtida e calibrada com base em 161 projetos cuidadosamente selecionados a partir de 2000 projetos candidatos.

Assim como seu antecessor, visa medir esforço, prazo, tamanho da equipe e custo exigido para o desenvolvimento, desde que se tenha como premissa, a sua dimensão. Conforme Reifer (1998) no COCOMO II não há determinações diretas de como medir o Custo, mas conhecendo-se prazo e equipe de trabalho, pode-se chegar a tal valor. Antes de ser usado, o modelo deve ser calibrado a partir dos dados históricos de projetos semelhantes àquele que se deseja estimar.

O COCOMO II possui 22 parâmetros, sendo 5 com efeito exponencial e 17 com efeito linear, que permitem ajustar o modelo às características de um projeto específico, conforme USC (2005), onde podem ser encontrados maiores detalhes dessa métrica, assim como a diferença entre as versões anteriores.

#### **2.4.8 – Análise Comparativa das Métricas de Software**

Conforme McGarry (2002), uma vantagem dos PF sobre as LOC é que podem ser obtidos logo no início do ciclo de vida, diretamente dos requisitos ou especificações, sendo úteis para estimativas realizadas no início do ciclo de vida independentemente de linguagem.

Aguiar (2003) conclui que de modo geral, os Pontos por Caso de Uso ainda não são uma opção aconselhável para as empresas, sendo mais conveniente utilizar os Pontos de Função, pelas seguintes razões:

- os PF são mantidos por uma organização internacional, o IFPUG, desde 1986 e possuem suporte no Brasil através do BFPUG (*Brazilian Function Point Users Group*);
- o IFPUG mantém um programa mundial de certificação profissional em PF, o qual confere o título de CFPS (*Certified Function Point Specialist*), realizado no Brasil pelo BFPUG;

- os PF são padronizados internacionalmente pela ISO, através da norma ISO/IEC 20926, possibilitando a uniformidade na aplicação;
- os PF modelam os requisitos a um nível de abstração mais elevado e independente dos artefatos do que os UCP, podendo ser utilizados por organizações que utilizem qualquer forma de representação dos requisitos, casos de uso ou outras;
- a existência de grande acervo de dados sobre Pontos de Função armazenados por diversas organizações possibilita a realização de estudos e comparações;
- a utilização dos Pontos de Função em contratos e licitações é uma realidade no Brasil, tendo surgido a partir da iniciativa de organizações governamentais e rapidamente alcançado o mercado em geral.

Conforme Abran (1999) o COSMIC-FFP, ainda não é tão disseminado no mundo quanto o APF do IFPUG, porém observa-se que muita pesquisa está sendo realizada.

Segundo Boehm (2000), O COCOMO busca medir esforço, prazo, tamanho de equipe e custo necessário para o desenvolvimento do software, desde que se tenha a dimensão do mesmo, através de estimativa de tamanho de software como Linhas de Código, Pontos por Função ou Pontos por Caso de Uso. Devido à idade dos projetos que embasaram o modelo anterior, assim como sua incapacidade de lidar com ciclos iterativos de desenvolvimento, o COCOMO 81, é atualmente considerado obsoleto, tendo sido substituído pela sua versão II.

Comparando com a métrica de Pontos por Função, Kemerer (1987) sugere que o modelo COCOMO pode ser mais adequado para ambientes de desenvolvimento não convencionais, como sistemas científicos e aplicações técnicas complexas, enquanto a PF tem características mais apropriadas para aplicações comerciais em geral.

## **2.5 - PMI (PROJECT MANAGEMENT INSTITUTE)**

### **2.5.1 – PMI no mundo**

Fundado em 1969, e sediado na Filadélfia, Pensilvânia, EUA, o Project Management Institute (PMI) é a principal associação internacional sem fins lucrativos em Gerenciamento de Projetos [Pmi (2006)]. Tem como principais objetivos:

- Avançar o estado da arte na prática da gestão de projetos;
- Fomentar o profissionalismo na gestão de projetos;
- Promover a aceitação da gestão de projetos como profissão e disciplina.

Ao final da década de setenta o PMI somava mais de 2.000 associados no mundo. Na década de oitenta, o número de associados do PMI continuou crescendo, bem como os programas e serviços oferecidos pela associação. Um Código de Ética foi adotado para a profissão e o primeiro Project Management Professional (PMP) foi certificado.

Em 1990, o PMI agregava mais de 8.500 associados e em 1993 este número crescia cerca de 20% ao ano. No início do século 21, o PMI tinha mais de 50.000 associados e mais de 10.000 Profissionais de Gerenciamento de Projeto (PMP) certificados em 37 países, divididos em 122 *chapters* que são sub-associações onde são discutidos temas mais específicos. Estavam em circulação mais de 270.000 cópias do PMBOK Guide.

### **2.5.2 - PMI no Brasil**

O Brasil foi o primeiro país a constituir um escritório do PMI (*Chapter*) fora dos Estados Unidos, no início da década de 80 [Pmisp (2006)]. Apesar do interesse já existente na época e de um crescimento significativo do número de associados, o *Chapter* do Brasil foi destituído em 1984.

Com a nova diretriz de expansão internacional do PMI e o avanço do Gerenciamento de Projetos no Brasil, no final dos anos 90 houve uma nova iniciativa para o estabelecimento de uma entidade nacional voltada para o tema. As dimensões continentais do país levaram o PMI a incentivar a criação de *Chapters* por estado da nação, a fim de manter o ideal de congregação dos profissionais.

O primeiro escritório a se estabelecer no Brasil nesta nova fase foi o de São Paulo, em 1998. No dia 13 de junho desse ano foi realizado o primeiro exame para *PMP*, contando com a participação de 33 profissionais. Atualmente no Brasil, o PMI está estabelecido no Distrito Federal e nos seguintes estados brasileiros: São Paulo, Minas Gerais, Paraná, Rio de Janeiro, Rio Grande do Sul, Amazonas, Santa Catarina, Goiás, Ceará, Espírito Santo, Bahia e Pernambuco [Pmibr (2006)].

### **2.5.3 - PMBOK**

O PMBOK (A Guide to the Project Management Body of Knowledge) [PMBok (2004)] representa todo o somatório de conhecimento dentro da profissão de Gerência de Projetos. Como em qualquer outra profissão esse conjunto de conhecimentos baseia-se nas contribuições daqueles profissionais e estudantes que aplicam esses conhecimentos no dia-a-dia. Nesse conjunto estão incluídos os conhecimentos já aprovados através de práticas tradicionais que são amplamente utilizadas, assim como, os conhecimentos de práticas mais inovadoras e avançadas que tem tido uma aplicação mais limitada.

O PMBOK também pretende fornecer uma terminologia comum, dentro da profissão e práticas, para a linguagem oral e escrita sobre Gerência de Projetos. Esse documento é destinado a qualquer profissional interessado na profissão de Gerência de Projetos, tais como: executivos seniores, gerentes de projeto, clientes e outras partes envolvidas no projeto, gerentes funcionais que tem colaboradores alocados às equipes de projetos, professores que atuam em disciplinas afins, etc.

O PMI se propõe a editar uma nova edição do Guia com as melhores práticas em gerenciamento de projetos a cada quatro anos, tendo lançado no final de 2004 sua terceira edição.

A 3ª edição do PMBOK recomenda 44 processos divididos em nove Áreas de Conhecimento, como podem ser visto na Figura 2.2, e em cinco Grupos de Processos, que independem das áreas de conhecimento às quais pertençam. Isso não significa que todos os processos descritos, devam ser sempre aplicados de forma uniforme em todos os projetos. A determinação dos processos adequados e o grau de rigor adequado na execução de cada atividade, devem ser avaliados e planejados pelo gerente do projeto.





FIGURA 2.2 – ÁREAS DE CONHECIMENTO DO PMBoK

#### 2.5.3.1 - Áreas de Conhecimento do PMBOK

- Gerenciamento do Escopo do Projeto - A preocupação fundamental do gerenciamento de escopo compreende em definir e controlar o que está ou não no projeto, de forma que este seja concluído de forma bem sucedida;
- Gerenciamento do Tempo do Projeto - Inclui todos os processos necessários para assegurar que o projeto seja implementado no prazo;
- Gerenciamento do Custo do Projeto - Inclui todos os processos necessários para assegurar que o projeto seja concluído dentro do orçamento aprovado;
- Gerenciamento da Qualidade do Projeto - Visa assegurar que o projeto irá satisfazer as necessidades para as quais foi empreendido;
- Gerenciamento de Recursos Humanos do Projeto - Inclui os processos para tornar mais efetivo o uso dos recursos humanos no projeto;
- Gerenciamento das Comunicações do Projeto - Contempla os processos para garantir a regular e apropriada geração, coleta, disseminação, armazenamento e descarte final das informações do projeto;
- Gerenciamento de Riscos do Projeto - É um processo sistemático de identificar, analisar e responder aos riscos do projeto;

- Gerenciamento de Aquisições do Projeto - Inclui os processos necessários à obtenção de bens e serviços externos à organização executora;
- Gerenciamento de Integração do Projeto – Visa assegurar que os diversos elementos do projeto sejam adequadamente coordenados.

#### **2.5.3.2 - Grupos de Processos**

- Iniciação – São processos que definem e autorizam o projeto ou uma de suas fases. Nela é nomeado e empossado o Gerente do Projeto;
- Planejamento – Refina os objetivos, definindo e planejando as ações necessárias para atender o escopo planejado para o projeto;
- Execução – Integra recursos materiais e pessoais para a realização do Plano de Gerenciamento do Projeto;
- Monitoramento e Controle – Monitora o progresso para identificar variações em relação ao que foi planejado, permitindo a tomada de ações corretivas;
- Processos de Encerramento – Conduz o projeto ao seu término, formalizando a aceitação do produto ou serviço do projeto.

#### **2.5.4 – Certificação PMP - Project Management Professional**

A certificação PMP é a credencial mais reconhecida no mundo para esta profissão. Para obtê-la, o profissional deve satisfazer os requisitos de educação e experiência além de se comprometer com o código de ética [Pmisp (2006)].

O certificado PMP tem validade de 3 anos, tendo que nesse período satisfazer os requisitos do PDP – Professional Development Program para manter o certificado. Deve enviar relatório de atividades ao PMI em até 3 anos a partir da certificação, comprovando reunir no mínimo 60 PDU's – Professional Development Units. São contabilizadas como PDU's a participação em cursos, atividades comunitárias e associativas, palestras, moderador ou membro de comitês, dirigente do PMI ou atividades profissionais na área.

## **2.6 - CMMI (CAPABILITY MATURITY MODEL INTEGRATION)**

### **2.6.1 - Introdução**

Equipes de desenvolvimento de software podem utilizar o CMMI para otimizar o tempo de desenvolvimento evitando retrabalho e aumentando a qualidade final do software produzido [SEI (2006)].

É o mais recente modelo de maturidade do SEI (Software Engineering Institute), um dos maiores influenciadores em gestão de processos de software em todo o mundo. Derivado dos modelos SW-CMM (*CMM for Software*) e SE-CMM (*CMM for System Engineering*), surgiu da percepção de que software básico e aplicações têm desenvolvimento em contextos integrados.

O modelo CMMI faz uma distinção não presente no modelo SW-CMM pois separa a Gerência de Requisitos (*Requirements Management*) da Elaboração de Requisitos (*Requirements Development*). Essa divisão pode ser considerada artificial, considerando a constante evolução dos requisitos, mas ajuda a separar a preocupação com o gerenciamento da preocupação com o entendimento e definição dos requisitos.

### **2.6.2 – Níveis de Maturidade**

O CMMI, como seus antecessores, não define como o processo deve ser implementado, mas prescreve suas características estruturais e semânticas em termos de objetivos e grau de qualidade. Mantém a mesma estrutura de 5 níveis de maturidade dos modelos anteriores, com caracterização bem distinta:

- **NÍVEL 1 (INICIAL)** – Não existe documentação e não há mecanismos de controle que permitam ao gerente saber o que está acontecendo, identificar problemas e riscos e agir de forma adequada. Não existem procedimentos padronizados, estimativas de custos e planos de projeto.
- **NÍVEL 2 (GERENCIADO)** - O planejamento e o controle de projetos são estabelecidos e definidos na forma de processos. É possível repetir o sucesso de um projeto anterior em outros similares. A organização deve conter controles básicos de projeto, incluindo o gerenciamento de projetos e a instituição de controles que assegurem que mudanças no projeto não destruam o que já foi feito. A organização está em condições de ter maior

controle sobre seus projetos e as estimativas são mais precisas, já que se desenvolve uma base histórica na qual se baseará, porém caso a empresa enfrente o desafio de desenvolver projetos de características diferentes das que está acostumada, usando uma tecnologia nova, por exemplo, esta informação será irrelevante, e a empresa poderá regredir ao nível 1.

- **NÍVEL 3 (DEFINIDO)** - Tanto as atividades de gerenciamento quanto de engenharia do processo de desenvolvimento de software estão documentadas, padronizadas e integradas em um padrão de desenvolvimento da organização. É necessário que a empresa introduza uma Metodologia de Desenvolvimento formal com um ciclo de vida definido e acompanhada de métodos, técnicas e ferramentas apropriadas, como inspeções e técnicas abrangentes de teste.
- **NÍVEL 4 (QUANTITATIVAMENTE GERENCIADO)** - Focado na qualidade do produto e do processo, leva a organização à condição de ser capaz de planejar a qualidade a partir da medição de indicadores, tanto para produtos quanto para processos. A empresa deve estabelecer métricas de forma a medir características específicas dos produtos.
- **NÍVEL 5 (OTIMIZADO)** - Refere-se ao melhoramento contínuo do processo, a partir da realimentação sistemática dos processos com os indicadores colhidos nos controles, a partir do gerenciamento das mudanças tecnológicas. O melhoramento contínuo do processo é conseguido através de análise quantitativa dos processos e pelo uso de idéias e tecnologias inovadoras.

### **2.6.3 – Áreas Chaves de Processos**

A partir do nível 2, cada nível do CMMI possui Áreas Chaves de Processos que definem a classificação da Organização, como segue:

- Nível 2: Gerência de Requisitos; Planejamento de Projeto; Acompanhamento e Controle de Projeto; Gerência de acordos com fornecedores; Gerência da Qualidade de processo e produto; Gerência de Configuração; Medição e Análise.
- Nível 3: Foco no processo da Organização; Definição do processo da Organização; Treinamento Organizacional; Gerência Integrada de projeto; Gerência de Risco; Desenvolvimento de Requisitos; Solução Técnica;

Integração de Produto; Verificação; Validação; Análise de Decisão e Resolução.

- Nível 4: Desempenho do Processo Organizacional; Gerência Quantitativa do Projeto.
- Nível 5: Análise Causal e Resolução; Inovação e Melhoria Organizacional;

#### **2.6.4 – Certificação CMMI**

O objetivo de muitas empresas tem sido a obtenção de certificações CMMI para atender às exigências do mercado. Uma certificação CMMI evidencia que a empresa está em um determinado nível de maturidade, apurado por auditores que verificam as seguintes fontes de informação:

- Instrumentação aplicada ao processo;
- Entrevistas e questionários aplicados à equipe;
- Apresentações da equipe para os auditores;
- Documentação gerada pelo processo.

### **3 - TRABALHOS RELACIONADOS**

#### **3.1 – Introdução**

Diversos trabalhos já foram desenvolvidos com temas relacionados de alguma forma ao foco desta dissertação, dentre os quais, são apresentados de forma resumida os três a seguir, com uma breve comparação com a aplicação SAGEP – Sistema de Apoio à Gestão de Escopo, apresentada no próximo capítulo.

#### **3.2 - PRAXIS (Processo para Aplicativos Extensíveis Interativos)**

Em Carvalho (2001) é apresentada uma proposta de criação de uma ferramenta CASE, denominada Praxis Mentor, de apoio à utilização do processo Praxis em projetos desenvolvimento de software, fornecendo informações sobre cada fase do processo e auxiliando na sua utilização. Os módulos principais da ferramenta são: Consulta ao Processo, Composição da Equipe, Comunicação, Controle de Projeto, Proposição de Melhorias e Gestão de Problemas.

##### **3.2.1 – Visão Geral do Praxis**

O processo de desenvolvimento Praxis foi criado no Departamento de Ciência da Computação da Universidade Federal de Minas Gerais destinado a suportar projetos didáticos em disciplinas de Engenharia de Software de cursos de Informática. Aplica-se principalmente ao desenvolvimento de aplicativos gráficos interativos baseados na tecnologia orientada por objetos e abrange tanto métodos gerenciais como técnicos. A definição completa do processo pode ser encontrada em Paula Filho (2003).

Segundo Carvalho (2001), o Praxis Mentor é uma ferramenta visual com o objetivo de auxiliar os usuários do Praxis em todas as fases do processo:

- instruindo o desenvolvedor no uso do processo;
- fornecendo acesso fácil aos documentos e modelos utilizados em cada fase;
- facilitando a gestão do projeto e do processo.

As principais funções do Praxis Mentor estão relacionadas na Tabela 3.1.

Nº	Nome da função	Necessidades	Benefícios
1	Orientação on-line ao uso do Processo	- Conhecimento do processo e de sua forma de utilização.	- Capacitação do usuário no processo; - Agilidade na utilização do processo; - Diminuição dos erros no processo.
2	Acesso aos documentos e modelos gerados pelo projeto	- Organização dos artefatos do projeto; - Padronização de estrutura de pastas para projetos.	- Agilidade na recuperação de qualquer artefato do projeto; - Facilidade na realização de cópias de segurança do projeto.
3	Suporte Gerencial (Gestão do Projeto e Cálculo de métricas)	- Planejamento de Projetos; - Conhecimento das métricas do processo.	- Capacitação dos gerentes no processo; - Maior agilidade no planejamento de um projeto; - Maior facilidade no cálculo das métricas do processo; - Diminuição dos erros no cálculo de métricas.
4	PIP ( <i>Process Improvement Proposal</i> )	- Histórico sobre o uso do processo.	- Auxílio na personalização do processo; - Contribuição para a melhoria do processo.

TABELA 3.1 – FUNÇÕES DO PRAXIS MENTOR – Fonte: Carvalho (2001)

Segundo Carvalho (2001), a primeira versão do Praxis Mentor tem as limitações abaixo, sendo sugeridas suas implementações em versões posteriores:

- O Praxis Mentor não se propõe a garantir a consistência entre os dados do projeto e os artefatos do processo.
- O Praxis Mentor não faz integração com nenhuma ferramenta de Gestão de Configurações, com ferramenta de Modelagem nem com Editores de Texto.
- Atividades como backup e recuperação das bases de dados do sistema ficarão a cargo da administração de dados e não serão providas pelo Praxis Mentor.
- O Praxis Mentor não se propõe a oferecer ajuda on-line, sendo suprida pela funcionalidade de Consulta ao Processo.
- O Praxis Mentor não possui tolerância à falhas.

### **3.2.2 – Análise Comparativa**

Com relação ao SAGEP, o Praxis Mentor assemelha-se no objetivo de, através de uma ferramenta automatizada, fornecer um apoio à gestão do projeto, em especial nas funções de Controle de Projetos e Gestão de Solicitações. Possui funções diferenciadas, como a Gestão de Comunicações, Gestão de Problemas e Controle de atividades da equipe do projeto que não estão no escopo do SAGEP.

Conforme Carvalho (2001), a métrica utilizada no desenvolvimento do Praxis foi a Análise de Pontos de Função. O requisito do Praxis “Implementação de recursos que facilitem o cálculo de métricas do processo” não foi implementado na versão original, ficando recomendados para trabalhos futuros, tendo sido contemplado pelo SAGEP.

Outra diferença básica entre os dois projetos é o objetivo de apoio à tarefa de estimativa de prazos e custos através da utilização de dados coletados em funções de porte similares armazenados em Base Histórica por parte do SAGEP.

### **3.3 – SIGERE (Sistema de Gestão de Requisitos)**

Em Cordeiro (2002) é apresentada uma proposta de criação de uma ferramenta automatizada de suporte à gestão de requisitos, denominada SIGERE.

#### **3.3.1 – Visão Geral do SIGERE**

Fundamenta-se no modelo CMMI e nas normas ISO/IEC 12.207 e 15.504 e utiliza a metodologia orientada a processo proposta pelo AGIR - Ambiente de Gestão da Inteligência da Realidade na modelagem da área chave Gerenciamento de Requisitos. Define como proposta essencial do trabalho “*o detalhamento do processo de gerenciamento de requisitos*” e também propõe uma ferramenta automatizada para auxiliar tal atividade, desenvolvida na plataforma Web.

Propõe como Trabalhos Futuros, entre outros, a implementação de métricas para previsão de custo e esforço e a implantação de tratamento analítico das informações armazenadas pelo SIGERE para melhorias em projetos futuros.

A Tabela 3.2 apresenta as funções da Ferramenta proposta.



<b>CASO DE USO</b>	<b>DEFINIÇÃO</b>
Cadastrar o projeto para o qual se fará o gerenciamento de requisitos;	Cadastramento dos projetos da organização para os quais se fará o gerenciamento dos requisitos.
Cadastrar os stakeholders;	Cadastramento dos stakeholders da organização.
Cadastrar a competência dos stakeholders na organização;	Cadastramento das competências dos stakeholders necessárias aos projetos da organização.
Cadastrar a competência dos stakeholders no projeto;	Cadastramento das competências dos stakeholders necessárias aos projetos da organização.
Cadastrar os requisitos de sistema do projeto;	Cadastramento dos requisitos da baseline que se origina no processo Desenvolvimento de Requisitos, como diz o CMMI.
Cadastrar a dependência dos requisitos;	Cadastramento do relacionamento de dependência entre os requisitos. Apresenta-se na estrutura hierárquica, tipo pai-filho.
Cadastrar as atividades dependentes dos requisitos.	Cadastramento do relacionamento de dependência entre os requisitos e as atividades do projeto.
Cadastrar atividades	Cadastramento das atividades de gerenciamento de projeto e de engenharia de software executadas pela organização.
Solicitar Mudança	Solicitação de mudança nos requisitos, podendo significar uma alteração e/ ou exclusão e/ou a inclusão de novos requisitos.
Identificar Stakeholders Pertinentes	Permite selecionar os stakeholders que serão responsáveis por analisar a Solicitação da Mudança.
Analisar Solicitação de Mudança	Permite aos stakeholders responsáveis procederem a Análise da Solicitação de Mudança.
Elaborar Proposta de Mudança	Permite ao responsável pela atividade que proceda a Elaboração da Proposta de Mudança.
Avaliar Impacto da Mudança	Permite ao Líder proceder a Avaliação de Impacto da Mudança, podendo ser aprovada para implementação ou rejeitada.

TABELA 3.2 – FUNÇÕES DO SIGERE – Fonte: Cordeiro(2002)

### 3.3.2 – Análise Comparativa

O SIGERE assemelha-se no objetivo de, através de uma ferramenta automatizada, fornecer um apoio à gestão do projeto, tendo praticamente todas as suas funções contempladas pelo SAGEP, embora de maneira mais simplificada, diminuindo assim o custo do controle. Enquanto no SIGERE o controle é feito por requisito, o SAGEP controla as funções do sistema, através dos Use Cases definidos.

### **3.4 – GEMETRICS (Ferramenta para Gestão de Projetos e Métricas)**

Vavassori (2002) apresenta uma proposta de criação de uma ferramenta CASE de suporte à Gestão de Projetos distribuídos e Métricas, denominada GEMETRICS.

#### **3.4.1 – Visão Geral do GEMETRICS**

Essa ferramenta coleta dados sobre cada atividade planejada para o projeto e suas dependências e executa um algoritmo para calcular a data final do projeto, bem como as datas de início e término mais cedo e, datas de início e término mais tarde de cada atividade. Para visualizar o planejamento das atividades foi implementado um gráfico de barras na aplicação.

Utiliza a métrica Análise de Pontos de Função para dimensionar o projeto, e telas para facilitar o cálculo. A partir daí, emite diversos relatórios com recursos gráficos.

O escopo da ferramenta tem como principais funções:

- Cadastros básicos para viabilizar o gerenciamento, tais como, atividades, processo, alocação de recursos, gerente responsável, dados dos projetos, etc.;
- Implementar a métrica ponto de função;
- Implementar o gráfico de Gantt para acompanhamento do projeto;
- Viabilizar o gerenciamento de aplicativos desenvolvidos;
- Implementar um conjunto de relatórios fornecendo dados sobre os projetos.

Alguns recursos existentes nessa ferramenta são específicos para o gerenciamento distribuído, bastando serem especificados os vínculos entre as atividades dos diferentes projetos.

As Funções do GEMETRICS, conforme Vavassori (2002), são:

- Cadastros de Aplicativo, projeto, recurso, gerente e sub-atividade;
- Alocação de Recurso;
- Identificação de funções do tipo dado
- Cadastro de modelo de processo de desenvolvimento e fases do processo;
- Identificação de funções do tipo transação;
- Informar fator de ajuste;

- Relatórios de custos por cargo e por fase, perfil profissional por fase, esforço por fase e referente ao tamanho dos sistemas;
- Relatórios comparativos do tempo e custo de um ponto de função entre projetos, da quantidade de pontos de função entre projetos e dos fatores de ajustes entre projetos;
- Estimativas do Aplicativo;
- Interfaces para configurações e para acompanhamento das atualizações.

### **3.4.2 – Análise Comparativa**

Assim como os demais trabalhos apresentados, um ponto em comum com o SAGEP é o apoio à Gestão de Projetos através de uma ferramenta automatizada, tendo porém um escopo bastante diferenciado. Em comum também tem a função de Calculadora de Pontos de Função, métrica adotada por ambos os trabalhos.

É integrada com ferramentas de gestão e explora recursos gráficos que facilitam a visualização dos dados comparativos. Contempla gestão de Recursos, Custos por etapa do processo, e integração entre projetos, que não estão no foco do SAGEP. Em contrapartida a função de apoio às estimativas de prazo e custo constantes no SAGEP, não está no escopo do GEMETRICS.

## **4 - ARQUITETURA – Sistema de Apoio à Gestão do Escopo do Projeto (SAGEP)**

### **4.1 – Introdução**

O Sistema de Apoio à Gestão do Escopo do Projeto – SAGEP visa dar suporte às funções inerentes ao cargo do Gerente de Projeto (GP) não pretendendo substituí-lo, mas auxiliá-lo com informações históricas referentes a tarefas similares desenvolvidas anteriormente, orientando as estimativas da duração e do custo das atividades do projeto.

O dimensionamento das funções utilizadas se faz necessário para evitar distorções nas pesquisas nas bases de dados históricos, evitando assim tratar de forma idêntica as funções bastante pequenas e normalmente simples de serem confeccionadas e as funções maiores e mais complexas. A métrica utilizada para o dimensionamento das funções foi a Análise de Pontos de Função, porém se houver preferência por outra métrica igualmente confiável, o SAGEP adequa-se facilmente, considerando porém, que desde o início da construção da base de dados histórica seja utilizada a mesma métrica.

Uma característica do SAGEP é a auto-alimentação de dados para cada projeto que vier a utilizá-lo, pois em uma de suas funções, os dados reais são registrados e além de permitir uma visão da margem de erro da estimativa em comparação com os dados reais, esses dados passarão a compor a base histórica, objetivando melhorar cada vez mais os valores de produtividade que serão consultados em projetos futuros.

Permite ainda um acompanhamento mais efetivo da situação do projeto e previsões referentes ao prazo e custo. Atribui ao *Sponsor* a responsabilidade de assumir alterações nessas previsões, aprovando ou não as solicitações de alteração do escopo, embasado em análises e justificativas registradas anteriormente. Com isso, o não cumprimento de prazo e o aumento do custo previsto não necessariamente serão assumidos pela equipe desenvolvedora.

## 4.2 - Requisitos do Sistema

A seguir são descritos os principais requisitos do SAGEP:

- O sistema deve proporcionar ao GP, auxílio na estimativa da duração do projeto, visando maximizar o índice de acerto;
- O sistema deve auxiliar o GP na estimativa dos custos, em especial de mão de obra, para o desenvolvimento do projeto;
- O sistema deve gerenciar as mudanças solicitadas por qualquer *stakeholder*, submetendo-as à aprovação ou rejeição do *Sponsor* do projeto, após análise do impacto da mesma pela equipe de desenvolvimento;
- O sistema deve proporcionar a todos os *stakeholders*, uma visão clara da situação atual do projeto, a qual deverá ser mantida atualizada periodicamente pelo GP;
- O sistema deve proporcionar a todos os *stakeholders* facilidades relacionadas ao controle do escopo do projeto, com um posicionamento quanto aos requisitos que o compõem.

## 4.3 – Segurança Lógica e Física do Sistema

O acesso ao SAGEP é controlado por um Sistema de Segurança de Acesso, onde o Administrador do Sistema define o perfil de cada usuário associando-o com os níveis de acesso permitidos.

Ao entrar na rede, o usuário digita sua chave de acesso e senha. As funções não autorizadas para esse usuário aparecem no menu, porém com cores diferenciadas identificando-as como não disponível.

Todas as atualizações efetuadas, registram internamente nas tabelas alteradas, a chave do usuário e a data e horário de atualização.

Com relação à proteção física dos dados, todo o ambiente no qual foi instalado tem um sistema de segurança física que protege os programas e os dados cadastrados em caso de incidentes.

No caso de implantação em outra instalação é recomendada a análise da necessidade ou não de um sistema de segurança física.

## **4.4 - Funções do SAGEP**

### **4.4.1 - Funções de Apoio ao Sistema**

- Cadastramento do Projeto
- Cadastramento dos *Stakeholders* da empresa
- Cadastramento de Tipos de Recursos de desenvolvimento
- Atualização do Custo de Mão de Obra dos recursos
- Associação do *Stakeholder* com o Projeto
- Cadastramento do Escopo Inicial do projeto
- Consulta Use Cases de um projeto
- Consulta dados de um Use Case especificado

### **4.4.2 - Funções de Gestão de Solicitações de Mudanças**

- Solicitação de Mudança no projeto
- Análise da Solicitação de Mudança
- Proposta de Criação, Alteração ou Exclusão de Use Cases
- Estimativa do Impacto da Solicitação no desenvolvimento do projeto
- Aprovação ou Rejeição de uma Solicitação de Mudança
- Consulta Solicitações de Mudanças de um projeto
- Consulta Dados de uma Solicitação de Mudança especificada
- Consulta Use Cases afetados por uma Solicitação de Mudança

### **4.4.3 – Funções de Apoio às estimativas de Prazo e Custo**

- Histórico comparativo de Use Cases de tamanho similar ao analisado
- Consulta detalhamento de Use Cases considerados similares ao analisado
- Consulta Recursos utilizados em um Use Case
- Cálculo de Pontos de Função
- Apuração e registro de dados reais no desenvolvimento de um Use Case

### **4.4.4 – Funções de Posicionamento de progresso do projeto**

- Avaliação do andamento do projeto
- Situação de um projeto

## 4.5 - Descrição do Sistema

As funções do sistema serão apresentadas a seguir, sendo comentadas algumas de suas particularidades. A Documentação Técnica do Projeto, contendo os Requisitos Funcionais e Requisitos Não Funcionais definidos pelos usuários, o Diagrama e Detalhamento dos Use Cases e o Diagrama de Transição de Estado constam no Apêndice C.

A Figura 4.1 apresenta um fluxo das atividades do SAGEP, onde as setas pretas mostram uma seqüência natural de execução das funções, as vermelhas mostram as atualizações nas Bases de Dados e as azuis mostram a origem dos dados consultados.

Uma ou mais solicitações podem ser registradas no sistema, quando houver a necessidade de mudança em requisitos definidos no escopo do projeto. Essas mudanças poderão representar alterações em requisitos já existentes ou mesmo novos requisitos que alterarão o escopo do projeto. Em qualquer situação, poderá haver um impacto na duração ou custo do projeto. Essa mudança somente será incorporada no escopo caso o Sponsor do projeto dê o seu aval. Para auxiliar tal decisão, o Gerente do Projeto fará uma análise do impacto. Caso a solicitação seja aprovada, as variações na duração e custo orçado do projeto serão automaticamente assumidas, tirando o ônus do atraso da equipe do projeto.

Caso na análise da solicitação, esta seja considerada inviável tecnicamente, pode ser cancelada, ficando porém registrada no sistema com a justificativa do motivo de cancelamento.

### 4.5.1 – Início de um projeto

Ao iniciar um projeto, o gerente do projeto deverá cadastrá-lo no SAGEP informando a data de início, previsão de término e custo total estimado. Esses dados devem ser levantados na fase de Anteprojeto, a qual já deve estar concluída, com as devidas aprovações do *Sponsor*. Essas informações serão apresentadas em telas de acompanhamento do projeto, indicando sua situação em relação ao planejamento.

Nesse momento também serão cadastrados os *stakeholders* do projeto, servindo de base para a liberação de acesso às funções do sistema correspondentes à sua atribuição no projeto.

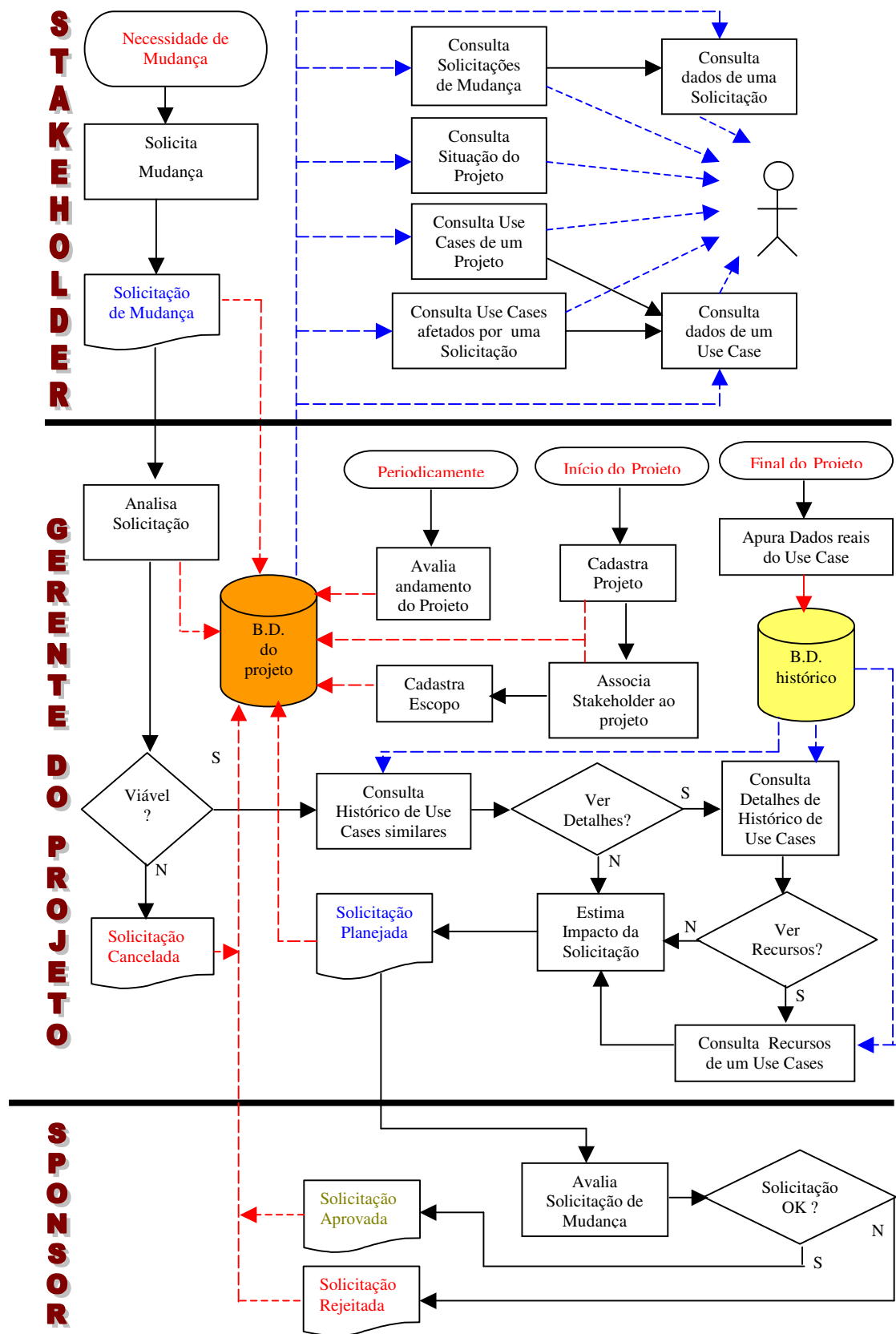


FIGURA 4.1 – DINÂMICA DE FUNCIONALIDADE DO SAGEP



#### 4.5.2 - Cadastramento do Escopo Inicial do projeto

O escopo aprovado para o projeto, será então cadastrado através da tela apresentada na Figura 4.2. O código de cada Use Case é gerado sequencialmente pelo sistema e seu nome e descrição serão informados. Nesse momento, o GP deve dimensionar essa função e estimar os recursos necessários e respectivas quantidades de horas de esforço para seu desenvolvimento.

**Escopo Inicial**

Cód.Projeto: 0519 Nome: CONTROLE DE AMOSTRAS NOS LABORATORIOS DA QUALIDADE

Cód.Use Case: 047 Nome: RELACIONAR PEDIDOS COM SALDO A PLANEJAR

Descrição do Use Case:  
INFORMAR QUAL A TONELAGEM QUE O PC APLICOU A UM DETERMINADO ITEM DE PEDIDO PARA FACILITAR A SELEÇÃO NA APLICAÇÃO.

Estimativas

Pontos de Função: 21

Consulta Histórico

Recurso	Horas Esforço (h.h.)	Valor Mao Obra Recurso
→ ANALISTA DE SISTEMAS	28	795,48
→ PROGRAMADOR	40	857,60
→		

NUM

FIGURA 4.2 – TELA DE CADASTRAMENTO DO ESCOPO INICIAL DO PROJETO

O sistema calcula e apresenta o valor da mão de obra de cada recurso informado, com base em valores parametrizáveis no sistema. Nesse momento o GP pode utilizar funções de apoio oferecidas pelo sistema para ajudá-lo a estimar o tamanho da função e a quantidade de horas necessárias de cada recurso.

#### 4.5.3 – Necessidade de Mudança

Durante o desenvolvimento do projeto, havendo necessidade de alteração significativa em suas funções ou mudanças no escopo já definido e aprovado pelo *Sponsor*, o solicitante deverá registrar a alteração pretendida, identificando-se e descrevendo a solicitação e sua justificativa na tela apresentada na Figura 4.3. O sistema identifica a Solicitação com um número sequencial à última registrada.

FIGURA 4.3 – TELA DE SOLICITAÇÃO DE MUDANÇA EM UM PROJETO

Essa solicitação não será assumida diretamente. Para tal será efetuada uma análise pela equipe de desenvolvimento coordenada pelo GP, o qual deverá preencher uma síntese dessa análise na tela referente à Figura 4.4.

FIGURA 4.4 – TELA DE ANÁLISE DE UMA SOLICITAÇÃO DE MUDANÇA

Se considerada inviável por discordância com diretrizes organizacionais ou mesmo mostrar ser inoportuna sua realização durante o desenvolvimento do projeto, o GP aciona o botão “Cancelar Solicitação” e confirma a decisão em uma Caixa de Diálogo que será apresentada pelo sistema. A Solicitação permanece registrada no sistema, porém com a situação “Cancelada” não sendo direcionada para o *Sponsor* avaliar. Qualquer que seja o motivo que tenha ocasionado o cancelamento da solicitação, deve ser anteriormente apresentado e consensado com o solicitante, evitando conflitos entre os *stakeholders* do projeto.

Se considerada viável, poderá ocasionar uma ou mais inclusões, exclusões ou alterações em Use Cases definidos, sendo que para registrá-las o GP acionará o botão correspondente na tela. Cada nova função indicada será dimensionada e terá os recursos estimados de forma similar ao cadastramento do escopo, o mesmo ocorrendo com as funções que sofrerão alteração.

Através da tela apresentada na Figura 4.5, o GP seleciona o Use Case a ser estimado e os recursos que serão envolvidos, informando a quantidade de horas de esforço previstas para cada recurso. O sistema apresenta o custo da mão de obra de cada recurso informado.

**Solicitação**

Cód.Projeto: 0519    Num.Use Case: 096    Num.Solic.: 001

Recurso	Horas	Mão de Obra (R\$)
→ ANALISTA DE SISTEMAS	24	681,84
→ PROGRAMADOR	48	1.029,12
→		
<b>Total:</b>	72	1.710,96

Dias Úteis Adicionais: 5    Custo Adicional em R\$ (Exceto Mão de Obra): 1

FIGURA 4.5 – TELA DE ESTIMATIVA DE IMPACTO DE UM USE CASE NO PROJETO

O total de horas informado não pode ser assumido como tempo adicional ao projeto, pois cabe ao GP identificar a possibilidade de utilizar mais de um recurso de mesmo tipo simultaneamente ou aproveitar um recurso ocioso durante a realização de outras atividades que pertençam ao Caminho Crítico do projeto. Após essa definição, é preenchida a quantidade de dias adicionais prevista e o custo adicional, sem considerar a mão de obra, caso seja aprovada a solicitação de mudança.

Após o dimensionamento e estimativas de todos os Use Cases que serão necessários para atendimento da solicitação de mudança, o sistema terá condições de calcular o impacto total da solicitação.

Através da tela apresentada na Figura 4.6, o *Sponsor* recebe as informações necessárias para decidir pela aprovação ou rejeição da solicitação. Além do nome do solicitante e as descrições da solicitação, justificativa e análise, a tela apresenta as previsões de data de implantação e custo total do projeto. Apresenta ainda o impacto da solicitação que está sendo avaliada, através da soma dos dias de acréscimo e variação de custo, englobando aqui também o custo da mão de obra referente às horas de esforço estimadas, para cada Use Case relacionado à Solicitação. Projeta também a nova data prevista para a implantação e o Custo total do projeto, caso seja definida a aprovação da solicitação.

Previsão Atual de implantação	Previsão de Custo Atual	Dias Úteis Adicionais	Variação do Custo	Nova Data prevista de implantação	Custo total do projeto (R\$)
14/02/2006	36.421,92	10	3.421,92	28/02/2006	39.843,84

FIGURA 4.6 – TELA DE AVALIAÇÃO DE UMA SOLICITAÇÃO

O *Sponsor* após analisar os dados apresentados, opta pela Aprovação ou Rejeição da Solicitação, escolhendo o botão correspondente para ser pressionado. Caso seja rejeitada, o sistema apresenta uma Caixa de Diálogo, onde o *Sponsor* deixará registrado o motivo da rejeição. Se for aprovada, o sistema alerta que com essa decisão o projeto estará assumindo automaticamente a nova data de previsão de implantação e o Custo Total informados, solicitando uma confirmação da aprovação, permitindo ainda que a mesma seja revista.

#### **4.5.4 – Apoio ao cálculo de Pontos de Função**

Para auxiliar na tarefa de dimensionamento de um Use Case, tanto na fase de previsão do escopo inicial e mudanças solicitadas quanto na apuração do seu tamanho real, após a conclusão do mesmo, o SAGEP apresenta uma calculadora de pontos de função.

Utiliza a regra internacional regulamentada pelo IFPUG, como pode ser aprofundado em IFPUG - International Function Point Users Group - Function Point Counting Practices Manual (2005). Os dados utilizados para o cálculo de Pontos de Função de um Use Case ficam registrados para facilitar possíveis necessidades de correções e mesmo facilitar o cálculo do real, baseado no previsto.

Essa função está disponível em 8 abas, sendo que na Figura 4.7, é apresentada a aba “Arquivos Lógicos Internos” que é similar às abas “Arquivos de Interface Externa”, “Entrada Externa”, “Saída Externa”, “Cons.Externa – Entradas” e “Cons.Externa – Saídas”. Nelas são preenchidos os valores devidos nas matrizes apresentadas e, através da pontuação correspondente, vai sendo calculada a quantidade de Pontos de Função Não Ajustados. Após a conclusão do preenchimento, são preenchidos os fatores de ajuste de cada uma das 14 características apresentadas na tela “Fator de Ajuste”, também apresentada na Figura 4.7, onde os Pontos de Função são reajustados em até 35% a mais ou a menos. O resultado obtido pode ser verificado na última aba da tela.

Ao sair da calculadora, caso tenha sido acionada por uma das telas do sistema, o valor dos Pontos de Função são automaticamente transportados para a tela que a acionou e os dados são armazenados para possível consulta posterior.

**- AAGP0025 - Cálculo de Pontos de Função**

Cons. Externa - Entrada	Cons. Externa - Saídas	Fator de Ajuste	Cálculo - Pontos de Função
Arquivos Lógicos Internos	Arquivos de Interface Externa	Entrada Externa	Saída Externa

Cód.Projeto: 0519 Cód.UC: 096

☒ PF Previsto  
☐ PF Real

Qtde. Itens de dados

	1 a 19	20 a 50	51 ou mais
Qtde. Registros Lógicos			
1	1		
2 a 5			
6 ou mais			

Quantidade de PF: 7

**- AAGP0025 - Cálculo de Pontos de Função**

Arquivos Lógicos Internos	Arquivos de Interface Externa	Entrada Externa	Saída Externa
Cons. Externa - Entrada	Cons. Externa - Saídas	Fator de Ajuste	Cálculo - Pontos de Função

Cód.Projeto: 0519 Cód.UC: 096

☒ PF Previsto  
☐ PF Real

Característica	Nível de Influência (0 a 5)
Comunicação de dados	3
Funções distribuídas	0
Desempenho	3
Configuração de equipamento	2
Volume de transações	0
Entrada de dados On-line	5
Interface com o usuário	5
Atualização on-line	1
Processamento complexo	1
Reusabilidade	3
Facilidade de implantação	1
Facilidade operacional	0
Múltiplos locais	0
Facilidade de mudanças	1

Fator de Ajuste =  $(\sum(NI) * 0,01) + 0,65$ : 0,90

**- AAGP0025 - Cálculo de Pontos de Função**

Arquivos Lógicos Internos	Arquivos de Interface Externa	Entrada Externa	Saída Externa
Cons. Externa - Entrada	Cons. Externa - Saídas	Fator de Ajuste	Cálculo - Pontos de Função

Cód.Projeto: 0519 Cód.UC: 096

☒ PF Previsto  
☐ PF Real

Pontos de Função Não Ajustados (PFNA): 17

Fator de Ajuste (FA): 0,90

Pontos de Função Ajustados (PF = PFNA \* FA): 15

FIGURA 4.7 – TELAS DE REGISTRO E CÁLCULO DE PONTOS DE FUNÇÃO

#### 4.5.5 – Apoio à Estimativa de Prazo e Custo através do Histórico

O GP informa o Use Case em análise, na tela correspondente à Figura 4.8. O sistema busca na base de dados histórica de dados reais de Use Cases já concluídos deste e de projetos anteriores, os que têm tamanho igual ou aproximado à quantidade de Pontos de Função do Use Case informado.

[illegible]

FIGURA 4.8 – TELA DE CONSULTA AOS DADOS HISTÓRICOS DE USE CASES

A aproximação permite contemplar os use cases não exatamente de mesmo tamanho, mas de uma faixa de percentual definida pelo GP. O critério para o sistema considerar os Use Cases da base histórica como similares ou não ao informado está parametrizado, permitindo assim ao GP ajustar o percentual desejado da faixa de Pontos de Função a ser selecionada.

O SAGEP pesquisa e apresenta os recursos utilizados nos Use Cases similares e calcula a média de horas de esforço consumidas de cada tipo de recurso, apresentando o resultado, juntamente com o custo de mão de obra atualizado, correspondente ao produto da quantidade de horas apresentada multiplicada pelo valor da hora atual desse recurso.

Caso o GP deseje verificar os dados utilizados para o cálculo da média apresentada, poderá acionar o botão “Consultar Detalhes”, para ser apresentada a relação dos Use Cases encontrados na base de dados Histórica de dimensionamentos considerados similares. Além da identificação dos Use Cases, apresenta seus tamanhos e quantidades totais de horas de esforço consumidas, conforme apresentado na Figura 4.9.

**Solicitação Atual**

Cód.Projeto: 0519    Cód.Use Case: 096    Quant. P.F. prev.: 15

**Use Cases Similares**

Código Projeto	Cód. Use Case	Quantidade P.F. real	Horas esforço-real
0081	055	17	53
0081	056	17	53
0081	057	18	55
0451	049	12	36
0519	003	12	37
0519	006	15	49
0519	012	15	50
0519	019	15	48
0519	023	17	56
0519	024	13	43
0519	025	13	41

FIGURA 4.9 – TELA DE CONSULTA DETALHES DE HISTÓRICO

Se o GP desejar o detalhamento desse total de horas de esforço, relativo a cada recurso, a tela apresentada na Figura 4.10 apresenta os tipos de recursos



utilizados no desenvolvimento do Use Case, a quantidade de horas de esforço de cada recurso e o valor da mão de obra assumido na ocasião de sua realização.

[illegible]

FIGURA 4.10 – TELA DE CONSULTA AOS RECURSOS UTILIZADOS EM UM USE CASE

#### 4.5.6 – Posicionamento da Situação do projeto

Periodicamente o GP deve informar a situação do projeto através da tela apresentada na Figura 4.11.

Sistema de Apoio à Gestão de Projetos - [AAGP0015 - Avaliar Andamento do Projeto]

Cadastros

Manutenção

Consulta

Janela

Cód.Projeto

Nome do Projeto

Última Atualização

0519

CONTROLE DE AMOSTRAS NOS LABORATORIOS DA QUALIDADE

06/11/2005

Início do Projeto

Término Previsto

% Tempo Transcorrido

Custo Previsto (R\$)

Situação

01/06/2005

14/02/2006

61,24

36.421,92

EM ANDAMENTO

% Conclusão

Custo Acumulado (R\$)

Projeção de Término

65

25.000,00

30/01/2006

Situação Atual do Projeto

Normal

Tendência de Atraso

Atrasado

Análise da variação do custo

Dentro do Previsto

Análise da variação da duração

Folga de 15 dias considerada importante para cobrir possíveis incidências desfavoráveis.

FIGURA 4.11 – TELA DE AVALIAÇÃO DO ANDAMENTO DO PROJETO

Ao selecionar o código do projeto o sistema informa a data da última atualização, situação do projeto, datas de início e término e custo total previstos, calcula o percentual de tempo transcorrido desde o início do projeto em relação à data de término.

O GP informa o percentual das atividades já concluídas, de acordo com o acompanhamento das atividades da equipe, o custo já realizado, uma análise das variações de custo e duração do projeto e aciona o botão correspondente à situação atual do projeto. O sistema calcula e apresenta a projeção de data de término baseado no percentual informado.

Ao atualizar a tela, se a data de projeção apresentada for diferente da data de término prevista, o sistema pergunta se deseja assumir a data projetada como nova data de previsão de término, permitindo a opção de assumi-la ou manter a data anterior, buscando adaptar o ritmo de trabalho àquela data.

Os dados relativos a esse posicionamento podem ser acompanhados por todos os *stakeholders* do projeto, através de uma tela de consulta da situação do projeto.

#### **4.5.7 – Consultas Diversas**

Além das consultas disponibilizadas e comentadas nos itens anteriores relativas à Situação do Projeto e às pesquisas de produtividade em Use Cases similares, o sistema disponibiliza a todos os *stakeholders*, as seguintes consultas:

- “Use Cases de um projeto” – Permite consultar qualquer projeto encerrado ou não, registrado no SAGEP. Os Use Cases propostos para o atendimento a uma solicitação de mudança ainda não aprovada pelo *Sponsor* do projeto aparecerão em vermelho para destacá-lo dos que já incorporam o escopo.
- “Dados de um Use Case” – Além do nome e descrição, apresenta o tamanho em Pontos de Função e estimativas de esforço e custo previsto e real, do Use Case solicitado. Se ainda não tiver o dimensionamento previsto ou o real, apresenta a linha correspondente em branco.
- “Solicitações de Mudanças” – Permite consultar o número, descrição e situação das solicitações de qualquer projeto encerrado ou não registrado no SAGEP.

- “Dados de uma Solicitação” – Apresenta o solicitante, data da solicitação, descrição, justificativa, análise, situação, impactos no custo e duração e datas de análise e aprovação da solicitação. Permite ainda o acionamento direto da consulta de Use Case afetados pela solicitação.
- “Use Cases afetados por uma Solicitação” – Apresenta o código e nome de cada Use Case relacionado à solicitação indicada. Permite o acionamento direto da consulta aos dados de um dos Use Cases apresentados.

#### 4.5.8 – Final do Projeto

A apuração de dados reais de um Use Case concluído deve ser realizada ao encerrar o projeto ou após a implementação de um Use Case. O GP deve informar a quantidade de horas de esforço realizadas por tipo de recurso e atualizar o seu tamanho em Pontos de Função, conforme apresentado na Figura 4.12. O sistema calcula e apresenta o custo de mão de obra de cada recurso indicado e registra esses dados no Banco de Dados Históricos utilizado para as consultas relativas à produtividade da equipe, fazendo assim com que em um próximo projeto os dados deste sejam contemplados, aumentando ainda mais a qualidade da estimativa sugerida pelo sistema.

**Sistema de Apoio à Gestão de Projetos - [AAGP0014 - Apurar dados reais]**

Cadastros Manutenção Consulta Janela

**Requisito**

Cód.Projeto: 0519 CONTROLE DE AMOSTRAS NOS LABORATORIOS DA QUALIDADE

Cód.Use Case: 096 CONSULTAR AMOSTRAS RECEBIDAS EM UM PERÍODO INFORMADO

**Dados Reais**

Pontos de Função: 16

Recurso	Horas	Mao de Obra (R\$)
→ ANALISTA DE SISTEMAS	30	852,30
→ PROGRAMADOR	52	1.114,88
→		
→		
→		
<b>Total:</b>	<b>82</b>	<b>1.967,18</b>

NUM

FIGURA 4.12 – TELA DE APURAÇÃO DE DADOS REAIS DE UM USE CASE

## **5 – ESTUDO DE CASO**

### **5.1 – Introdução**

Utilizou-se como laboratório a Gerência de Desenvolvimento de Sistemas de uma empresa siderúrgica de grande porte do estado de São Paulo, a qual desenvolve sistemas para uso da própria empresa, em especial no planejamento, logística e controle operacional das fábricas.

Depois de décadas de experiência no desenvolvimento e manutenção de sistemas em mainframe, em meados dos anos 90 teve início a migração dos milhares de programas para a plataforma cliente/servidor, motivada pela preocupação com a chegada do ano 2000 e a validação dos sistemas existentes, alguns com mais de 25 anos, com relação aos erros em potencial causados pela utilização apenas da dezena final do ano, fenômeno mundialmente conhecido como “*bug do milênio*”.

Como estratégia adotada pela empresa, os sistemas das áreas administrativa, financeira e controle de estoque foram substituídos pelo SAP/R3 [Sap (2006)], enquanto alguns sistemas envolvidos com a produção foram migrados para substituir o anterior antes da chegada do ano 2000 e outros foram checados e testados para suportarem o “ano 00” sem impactos negativos.

A partir daí, manter o mainframe com poucos sistemas tornava-se economicamente inviável. Continuou-se então o programa de *downsizing*, objetivando o desligamento do mainframe, o que foi atingido em novembro de 2003.

A impossibilidade de adiamento da data limite para as migrações, a pressão para o esvaziamento do mainframe e a falta de experiência na nova plataforma, geraram grande desgaste na equipe e deixaram evidentes os problemas causados pela falta de um gerenciamento efetivo dos projetos.

### **5.2 – Situação encontrada**

Em julho de 2004 foi realizada uma pesquisa, cujo modelo encontra-se no Apêndice A, com o objetivo de identificar as principais causas dos problemas apresentados em projetos de software na empresa, tomando-se por base um universo de 36 projetos desenvolvidos nos quatro anos anteriores. Algumas das conclusões consideradas significativas estão descritas a seguir:

- Ocorreram atrasos em 31 projetos em relação às estimativas iniciais, correspondendo a 86% do total de projetos pesquisados, tendo como média de atrasos um tempo superior em 99,1% da duração prevista;
- Entre os que atrasaram, 65% tiveram alterações de requisitos por parte dos usuários, que comprometeram a data compromissada;
- Em 49% dos projetos pesquisados causou frustração em usuário, por falta de requisitos não definidos na fase de Conceituação, com a alegação de que consideravam que estava implícito no entendimento do escopo;
- Em 57% dos projetos pesquisados foram detectados erros após a implantação, em alguma função que sofreu alteração depois de concluída e testada.

Os dados da pesquisa comprovam o que diz Kotonya (1998) quanto à falta de um efetivo gerenciamento de requisitos que compromete o sucesso de muitos projetos e não são muito diferentes dos dados apresentados em Chaos (2003) sobre o desenvolvimento de milhares de projetos pesquisados, citados no Capítulo 1.

### **5.3 – Desenvolvimento da Ferramenta de Apoio**

Segundo Kotonya (1998), o ponto de partida para o sucesso do projeto é um bom levantamento de requisitos em sua fase inicial, envolvendo os principais stakeholders do projeto. O produto dessa fase, geralmente chamada de “Estudo Preliminar” ou “Anteprojeto”, deve ser formalmente documentado e aprovado pelo Sponsor, não podendo gerar dúvidas quanto ao escopo do projeto.

Os requisitos do SAGEP foram levantados através de uma reunião JAD – Joint Application Design, ocorrida em maio de 2005, da qual participaram profissionais da empresa, com experiência em desenvolvimento de sistemas, e funcionários de áreas usuárias de sistemas, conforme ata apresentada no Apêndice B.

A empresa em questão utiliza o ambiente cliente/servidor com Sistema Operacional Windows NT, Banco de Dados Oracle e linguagem Centura. O SAGEP foi desenvolvido nesse ambiente para fins de utilização experimental e obtenção dos resultados reais, mas a partir da definição do projeto apresentada no Apêndice C, pode ser migrado para outra linguagem ou plataforma. Suas principais funções e interfaces estão descritas no capítulo 4 deste trabalho.

Foram levantados os dados de 82 Use Cases referentes a 4 projetos concluídos nos últimos anos na referida empresa, contando com a colaboração dos profissionais responsáveis por tais desenvolvimentos para a recuperação das informações necessárias. Cada atividade foi dimensionada em seus dados reais através do SAGEP e serviram como carga inicial das tabelas históricas do sistema, possibilitando assim a utilização plena das funções do sistema. Na impossibilidade de se obter os dados para a referida carga inicial, o SAGEP poderia ser utilizado nos primeiros projetos nas funções de controle de escopo e solicitações de mudanças. Os dados obtidos nesses primeiros projetos referentes ao dimensionamento duração e custos das funções, devem ser armazenados normalmente e após alguns projetos pode ser utilizada também a função de apoio a tais estimativas análogas.

Após a carga de dados iniciais, oito projetos iniciados a partir de maio de 2005 serviram de pilotos para o SAGEP, sendo cadastrados no sistema e tendo as estimativas dimensionadas com seu auxílio.

Todos os stakeholders responsáveis pelo desenvolvimento desses projetos foram conscientizados quanto à metodologia adotada para a gestão do escopo, e receberam permissão de acesso para as consultas e registro de solicitações. Inicialmente a reação de diversos stakeholders foi negativa, por considerar a alteração de procedimento como aumento de burocracia e perceberem que as mudanças em relação ao que foi definido, que sempre tiveram facilidade de fazer, passariam a serem controladas e questionadas quanto às reais necessidades, sendo naturalmente rejeitadas aquelas que se configurassem apenas como desejo pessoal sem agregar valor ao projeto.

Após algumas reuniões de apresentação desse procedimento, onde foram ressaltados os ganhos pretendidos e os benefícios que as áreas clientes do projeto passariam a ter com as práticas de gerenciamento a serem adotadas, foram superadas algumas barreiras para a aceitação do novo procedimento.

Um fato que foi relevante para essa mudança de visão dos usuários, foi a estratégia de convocação dos gerentes dessas áreas clientes para as reuniões de apresentação, pois estes perceberam mais facilmente os benefícios pretendidos e ajudaram a convencer os contrapartes que em geral apóiam o desenvolvimento dos sistemas para suas áreas.

#### 5.4 – Análise de Resultados

Os tempos previstos e reais dos Use Cases utilizados na carga inicial foram obtidos nos cronogramas dos respectivos projetos e a partir da diferença entre a estimativa e o tempo realizado, foram calculados os percentuais de erro de tempo de análise e de programação, sendo os resultados apresentados na Tabela 5.1, considerando com sinal “-” as estimativas com tempos menores e com sinal “+” as estimativas com tempos maiores do que os realizados.

Use Case	Erro Prev. x Real		Use Case	Erro Prev. x Real		Use Case	Erro Prev. x Real	
	Análise	Progr.		Análise	Progr.		Análise	Progr.
1	- 40%	- 19%	29	- 21%	- 17%	57	- 33%	- 10%
2	- 50%	- 25%	30	- 42%	- 47%	58	- 15%	0%
3	- 42%	- 11%	31	- 33%	- 47%	59	- 11%	+ 8%
4	- 50%	- 25%	32	- 40%	- 54%	60	- 25%	0%
5	- 40%	- 31%	33	- 44%	- 33%	61	- 28%	- 17%
6	- 57%	- 35%	34	- 63%	- 67%	62	- 38%	- 20%
7	- 42%	- 6%	35	- 38%	- 33%	63	- 22%	- 4%
8	- 40%	- 25%	36	- 25%	- 33%	64	- 17%	- 17%
9	- 50%	- 25%	37	- 25%	- 33%	65	- 10%	- 3%
10	- 33%	- 11%	38	- 42%	- 40%	66	- 33%	- 11%
11	- 50%	- 45%	39	- 50%	- 40%	67	0%	+ 11%
12	- 33%	- 6%	40	- 38%	- 40%	68	- 38%	- 28%
13	- 40%	- 31%	41	- 58%	- 67%	69	- 38%	- 11%
14	- 40%	- 19%	42	- 54%	- 53%	70	- 44%	- 23%
15	- 50%	- 25%	43	- 54%	- 53%	71	- 27%	- 29%
16	- 42%	- 11%	44	- 33%	- 33%	72	- 25%	- 27%
17	- 25%	- 17%	45	- 17%	- 33%	73	- 17%	- 20%
18	- 25%	- 17%	46	- 10%	- 6%	74	- 15%	- 18%
19	+ 10%	+ 15%	47	- 14%	0%	75	- 18%	- 14%
20	- 19%	- 13%	48	- 6%	+ 9%	76	- 21%	- 27%
21	0%	+ 14%	49	- 17%	- 19%	77	- 29%	- 27%
22	- 18%	+ 7%	50	- 42%	- 25%	78	- 23%	- 29%
23	- 13%	- 14%	51	- 33%	- 25%	79	- 27%	- 29%
24	- 19%	0%	52	- 27%	+ 11%	80	- 22%	- 17%
25	- 21%	- 15%	53	- 28%	- 25%	81	- 22%	- 17%
26	- 22%	- 19%	54	- 44%	- 36%	82	- 25%	- 25%
27	- 20%	- 21%	55	- 28%	- 33%	Média	30,22 %	23,16 %
28	- 35%	- 34%	56	+ 7%	+ 10%			

TABELA 5.1 – ERROS DE DURAÇÃO DE ATIVIDADES (PREVISTO X REAL)

A análise dos resultados, mostra que a previsão de duração correspondeu ao realizado em apenas 2 Use Cases para análise e em 4 Use Cases para programação. Na grande maioria dos casos, a previsão ficou abaixo do real, ocorrendo em 78 Use Cases para a previsão da análise e em 70 Use Cases para a previsão de duração de programação. A média de erros na duração da análise foi de 30,22% e da programação foi de 23,16%, apresentando como maiores erros 63% para a estimativa de análise e 67% para a estimativa de programação.

Na fase de validação do SAGEP foram acompanhados 8 projetos, de forma experimental. Apresentaram uma média de atraso total do projeto de 8,6% em relação à duração prevista, bem inferior aos 99,1% apresentados nos projetos realizados em anos anteriores, conforme pesquisa citada no item 5.2 deste trabalho. Outro fato considerado significativo foi a constatação que de 27 solicitações de mudanças apresentadas pelos stakeholders após a definição do escopo do projeto correspondente, 12 foram rejeitadas pelos respectivos Sponsors para não comprometerem a data de implantação do projeto. Isso mostra que o que estava sendo solicitado não representava ganhos significativos para o projeto e sem a formalização e correspondente avaliação do Sponsor, as mesmas acabariam sendo realizadas mesmo sem agregar valor ao cliente.

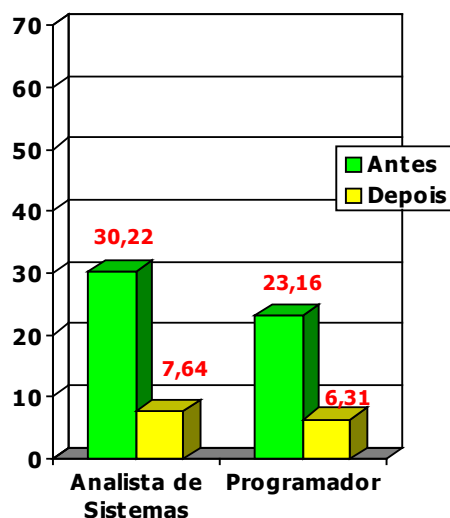
As estimativas de duração das funções dos projetos acompanhados são apresentadas na Tabela 5.2 e em comparação com os resultados apresentados na Tabela 5.1 a média de erro caiu de 30,22% para 7,64% no tempo previsto para análise de sistemas e de 23,16% para 6,31% nas horas previstas para programação. Quanto às maiores distorções entre os tempos previstos e os realizados, ocorreu também a queda de 63% para 22% para as horas de análise e de 67% para 20% nas horas de programação, conforme mostra a Figura 5.1. Em ambas tabelas, os percentuais de médias de erros foram calculados considerando como erro tanto a previsão à maior quanto à menor para evitar que o adiantamento de um projeto anulasse o atraso de outro.



Use	Erro Prev. X Real			Use	Erro Prev. X Real			Use	Erro Prev. X Real		
Case	Análise	Progr.		Case	Análise	Progr.		Case	Análise	Progr.	
83	-	7%	- 20%	115	+	7%	+ 6%	147		0%	+ 13%
84		0%	+ 5%	116		0%	0%	148	+	13%	0%
85	-	8%	- 7%	117	-	13%	- 14%	149	-	4%	- 6%
86		0%	0%	118		0%	- 11%	150	+	6%	- 9%
87	+	7%	0%	119	+	10%	- 7%	151		0%	+ 3%
88	-	8%	- 7%	120	-	20%	0%	152	+	13%	- 7%
89	-	4%	- 7%	121	+	15%	+ 14%	153	-	11%	- 8%
90	+	7%	0%	122	-	8%	- 7%	154	-	6%	- 5%
91	-	20%	0%	123		0%	+ 7%	155	+	8%	- 3%
92	-	8%	- 11%	124	+	9%	+ 7%	156	-	4%	- 7%
93	-	5%	+ 4%	125	-	13%	- 10%	157	-	11%	+ 9%
94		0%	- 3%	126	-	7%	0%	158	-	17%	- 7%
95	-	8%	0%	127		0%	+ 7%	159	+	4%	+ 7%
96	+	4%	- 6%	128	-	4%	+ 7%	160	-	10%	- 4%
97	-	15%	- 7%	129	-	4%	0%	161	-	8%	- 3%
98	+	9%	+ 15%	130	-	10%	- 6%	162		0%	+ 3%
99	-	11%	0%	131	-	8%	- 17%	163	+	4%	- 7%
100	-	10%	- 8%	132		0%	+ 3%	164	+	10%	+ 8%
101		0%	+ 3%	133	-	4%	- 7%	165	+	5%	- 7%
102	-	11%	- 8%	134	+	10%	+ 11%	166	+	8%	- 13%
103	+	9%	- 11%	135	+	5%	- 7%	167		0%	- 3%
104	-	22%	0%	136		0%	0%	168	-	11%	- 4%
105	-	11%	- 13%	137	-	11%	- 4%	169	+	6%	+ 8%
106		0%	+ 6%	138	+	6%	+ 8%	170		0%	+ 9%
107	+	8%	+ 13%	139	+	7%	+ 10%	171	-	5%	+ 4%
108		0%	- 5%	140	+	20%	+ 6%	172	-	13%	- 14%
109		0%	+ 9%	141	+	14%	+ 4%	173	-	14%	- 3%
110	-	17%	0%	142	+	14%	+ 10%	174	+	9%	+ 6%
111		0%	0%	143	-	11%	+ 8%	175	+	14%	+ 4%
112	+	5%	- 15%	144	+	5%	+ 7%	176	+	4%	- 6%
113	+	14%	+ 9%	145	-	10%	0%	177	+	10%	+ 8%
114	+	8%	0%	146	+	10%	- 7%	<b>Média</b>	-	<b>7,64%</b>	- <b>6,31%</b>

TABELA 5.2 – ERROS DE DURAÇÃO DE ATIVIDADES (PREVISTO X REAL) APÓS O SAGEP

**% erro médio nas funções**



**Maior % erro em uma função**

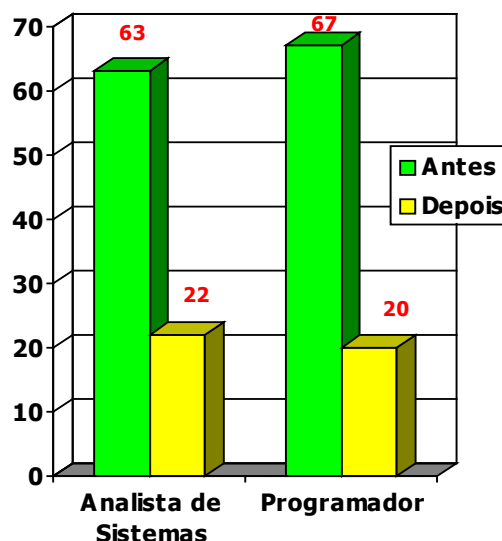


FIGURA 5.1 – ANÁLISE DE RESULTADOS

## 5.5 – Análise da Ferramenta SAGEP

Os resultados analisados mostram que o SAGEP mostrou-se eficaz no seu objetivo de auxiliar o Gerente de Projetos na gestão das solicitações de mudanças nos requisitos que delimitam o escopo do projeto dando suporte às estimativas dos impactos na duração e custos, com base no histórico de projetos anteriores de porte similar.

Como pré-condição pode-se destacar que para a utilização dessa ferramenta no apoio às estimativas análogas de tempo e custos, são necessários dados históricos referentes ao desenvolvimento de projetos semelhantes pela equipe desenvolvedora. Para a implantação em uma empresa nova, que não possua um histórico do desempenho dos desenvolvedores, durante os primeiros projetos são permitidas as utilizações das demais funções, como o controle dos requisitos que compõem o escopo aprovado para o projeto, o controle das solicitações pendentes, aprovadas e rejeitadas, etc. Devem ser registrados, porém, os resultados obtidos nesses primeiros projetos, para serem utilizados nas estimativas de futuros projetos;

Como limitação pode-se destacar que em caso de mudança de plataforma, linguagem ou técnicas de análise, deve-se considerar que as estimativas consultadas podem apresentar distorções no desempenho da equipe.

## **5.6 – Considerações Finais**

Em poucos meses de utilização do SAGEP, ficou visível a mudança na aceitação do procedimento adotado, por parte tanto dos Gerentes de Projetos quanto dos usuários e clientes, pois embora alguns passaram a ter menos autonomia nas alterações do projeto, perceberam que o nível de controle dos requisitos que delimitam o escopo e a diminuição de alterações no projeto é favorável a todos, pois impacta bem menos nos resultados esperados.

## **6 – CONCLUSÃO**

### **6.1 – Introdução**

Embora todas as áreas de conhecimento do Gerenciamento de Projetos sejam de suma importância, a Gestão do Escopo do Projeto merece atenção especial pois seu descontrole certamente impactará fortemente nas demais áreas de atuação.

As fases de levantamento e análise dos requisitos devem ser cuidadosamente executadas, delineando e documentando todo o escopo do projeto de forma clara e não ambígua. Toda mudança no escopo deve ser bem analisada e sempre que possível adiada para uma nova versão da aplicação. Infelizmente nem sempre isso é possível, pois mudanças na legislação, mudanças nos procedimentos da área de atuação do sistema e principalmente a identificação de falhas no levantamento e análise dos requisitos entre outros motivos, muitas vezes forçam que as mudanças ocorram. Se forem inevitáveis, resta ao Gerente do Projeto minimizar o impacto das mudanças no projeto em desenvolvimento. Para isso todos os stakeholders devem estar conscientizados de que é necessário formalizar toda e qualquer solicitação de mudança, pois a informalidade faz com que o descontrole domine o projeto.

### **6.2 – Conclusões**

- Todo projeto tem um “dono” e o Gerente do Projeto deve esforçar-se para que os compromissos quanto à duração combinada, o orçamento planejado e o escopo definido sejam cumpridos. Todas as mudanças no que foi inicialmente tratado devem ser devidamente avaliadas e aprovadas pelo Sponsor, que é o legítimo representante do cliente, considerando-se sua real necessidade. O estudo de caso demonstrou que ao se submeter à solicitação de mudança com o respectivo estudo de impacto à aprovação do Sponsor, acontece um filtro natural das solicitações, considerando-se o custo x benefício dessa mudança.
- Embora não sejam as únicas áreas que exigem cuidados do gerente do projeto, a Gestão de Escopo, Tempo e Custo compõem a restrição tripla do projeto e merecem atenção especial [Xavier (2005)]. Mudanças no escopo normalmente provocam alteração na duração e no custo do projeto. Para se estimar o impacto de tal mudança, o PMI recomenda a analogia com o

realizado em outros projetos semelhantes [PMBok (2004)]. A métrica utilizada no estudo de caso foi a Pontos de Função por atender às condições levantadas e serem mundialmente difundidas, o que facilita a comparação da produtividade alcançada entre desenvolvedores de diferentes empresas, mas o SAGEP não está restrito à sua utilização. Deve-se ter o cuidado, porém, de que toda a base histórica esteja com Use Cases dimensionados de forma idêntica, utilizando Pontos de Função ou outra métrica similar a ser adotada na empresa que utilizar este sistema.

- O SAGEP mostrou-se útil para auxiliar o Gerente no controle do Escopo do projeto, conforme mostram os resultados apresentados, atendendo os objetivos para o qual foi criado. A devida documentação do escopo, seu correto dimensionamento através de uma métrica que possa ser utilizada em tempo de definição e a formalização de todas as mudanças, para serem avaliadas e aprovadas ou não pelo Sponsor, permitiram um resultado melhor em relação aos projetos que não utilizaram essa ferramenta.

### **6.3 Sugestões para trabalhos futuros**

Este trabalho teve como proposta o auxílio na Gestão do Escopo do projeto, dando suporte para a tomada de decisão da aprovação ou não das mudanças solicitadas, a partir de estimativa de tempo de esforço de cada recurso necessário, baseada em dados reais coletados de funções semelhantes já concluídas. Com essa estimativa, o gerente do projeto fará uma análise das disponibilidades dos recursos necessários e estimará o possível atraso no término do projeto caso a mudança seja aprovada. Ficam como propostas para futuros trabalhos:

- A integração do SAGEP com ferramentas de gestão do tempo, e o desenvolvimento de funções de otimização dos recursos, para que o tempo de desenvolvimento do projeto possa ser mais bem administrado.
- O SAGEP pode ser evoluído também para ser integrado com ferramentas gráficas de gestão do escopo e planilhas de custo, que podem ser atualizadas automaticamente no momento da aprovação de uma mudança pelo Sponsor.
- Outras áreas do gerenciamento de projetos podem ser desenvolvidas e integradas ao SAGEP, como a Gestão de Riscos, onde o registro de

incidências de problemas pode servir de base para o indicador de probabilidade do risco.

- A Gestão da Qualidade pode também ser integrada com o SAGEP, integrando técnicas recomendadas pelo PMBoK e pelo CMMI, auxiliando a obtenção de certificações dos órgãos correspondentes.

#### **6.4 Considerações Finais**

A definição de procedimentos, mesmo sendo considerada muito importante para a melhoria da qualidade e exigida para a obtenção de certificações, [Chinelato (2004)], muitas vezes causa resistência por parte dos envolvidos. A formalização das solicitações de mudanças é fundamental para o correto funcionamento do SAGEP, devendo se necessário, haver uma preparação dos envolvidos, conscientizando-os da importância desse procedimento.

Outro procedimento indispensável é o registro de dados realizados para a alimentação do banco histórico, pois essas informações serão muito úteis nos próximos projetos que utilizarão o SAGEP.

## ANEXO A

### DETALHAMENTO DA ANÁLISE DE PONTOS DE FUNÇÃO (APF)

Segundo Abran (1999) , Dias (2003), Calazans (2005) e IFPUG (2005), a Análise de Pontos de Função é um método utilizado internacionalmente para a medição do software em desenvolvimento, visando estabelecer uma medida de seu tamanho, com base na funcionalidade a ser implementada, sob o ponto de vista do usuário sem se preocupar com a tecnologia que será utilizada na implementação. É composta de 3 etapas. Na primeira é calculado o total de pontos por função não ajustados. A seguir é determinado o valor do fator de ajuste e por último é calculado o total de pontos de função ajustados.

#### 1 – CONTAGEM DE PONTOS DE FUNÇÃO NÃO AJUSTADOS (PFNA)

Os pontos de função não ajustados refletem as funcionalidades fornecidas pelo sistema para o usuário. Leva em conta dois tipos de função, bem como sua complexidade.

FUNÇÕES DE DADOS (ALI / AIE) - Representam as funcionalidades relativas aos requisitos de dados internos e externos à aplicação. Ambos são grupos de dados logicamente relacionados ou informações de controle que foram identificados pelo usuário, tendo como diferença básica:

- Arquivo Lógico Interno (ALI) - É um grupo lógico de dados relacionados ou informação de controle identificado pelo usuário e mantido dentro da fronteira da aplicação. Sua intenção primária é manter os dados que sofrem manutenção através de um ou mais processos elementares da aplicação que está sendo contada. Exemplos de ALI's:
  - Tabelas de dados (sem existir porém uma relação de 1 tabela = 1 ALI)
  - Arquivos de configuração mantidos pela aplicação.
  - Arquivos de mensagens de erros, desde que mantidos pela aplicação.

- Arquivo de Interface Externa (AIE) - É um grupo lógico de dados relacionados ou informação de controle identificado pelo usuário apenas referenciado pela aplicação, mas mantido por outra aplicação.

O primeiro passo para a contagem das funções de dados consiste em identificar ALI's e AIE's, e classificá-los segundo sua complexidade funcional definida com base em dois conceitos: registros lógicos e itens de dados.

Registros Lógicos são subconjuntos de dados dentro de um ALI/AIE, que foram reconhecidos pelo usuário. Se o usuário não reconhecer subconjuntos de dados em um ALI/AIE, então se deve contar como um registro lógico.

Um Item de Dados é um campo reconhecido pelo usuário como único e não repetido, só devendo ser contados os itens utilizados pela aplicação que está sendo dimensionada.

Contando-se os registros lógicos e os itens de dados de um ALI/AIE individualmente, pode-se chegar à sua complexidade, utilizando a Tabela A.1.

Complexidade		QUANTIDADE DE ITENS DE DADOS		
		1 a 19	20 a 50	51 ou mais
Nº R E G.  L Ó G I C O S	Somente 1	Simples	Simples	Média
	2 a 5	Simples	Média	Complexa
	6 ou mais	Média	Complexa	Complexa

TABELA A.1 – COMPLEXIDADE DAS FUNÇÕES DE DADOS (ALI / AIE)

FUNÇÕES TRANSACIONAIS (EE / SE / CE) - Representam as funcionalidades de processamento de dados do sistema fornecidas para o usuário. A Figura A.1 apresenta uma visão geral dos tipos de funções que são considerados na contagem da APF.



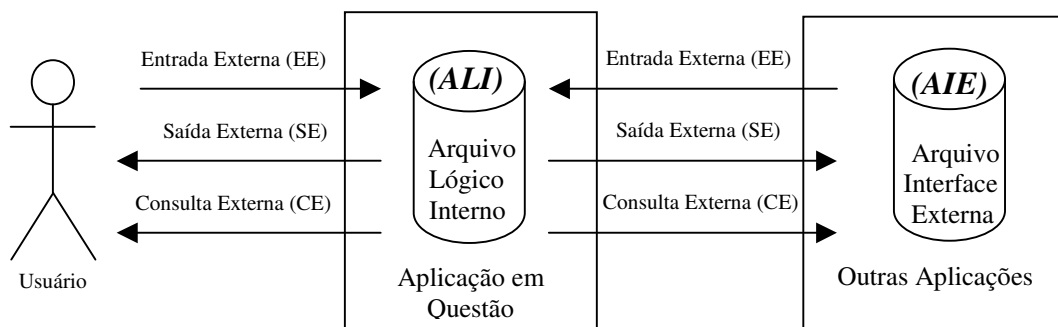


FIGURA A.1 – FUNÇÕES DE UMA APLICAÇÃO SEGUNDO A APF – Fonte: Falbo (2005)

- Entradas Externas (EE) - São processos elementares de dados (ou informações de controle) que entram pela fronteira da aplicação. O objetivo principal de uma EE é manter um ou mais ALI's ou alterar o comportamento do sistema. Identifica-se como uma função de entrada cada transação que tenha pelo menos:
  - Um formato diferente;
  - Uma lógica de processamento diferente;
  - Modifica arquivos (ALI) diferentes;
  - Os ALI's e AIE's referenciados são diferentes de outras EE's;
  - Diferentes operações sobre os mesmos arquivos.

Exemplos:

- Fluxos de dados que entram no sistema, gerados nas entidades externas;
- Documentos digitados;
- Leituras ópticas e magnéticas;
- Transações em disquetes e fitas.
- Saídas Externas (SE) - São processos elementares que enviam dados (ou informações de controle) para fora da fronteira da aplicação. Seu objetivo é mostrar informações recuperadas através de um processamento lógico (isto é, que envolva cálculos ou criação de dados derivados) e não apenas uma simples recuperação de dados. Uma SE pode, também, manter um ALI ou alterar o comportamento do sistema. Identifica-se como uma função de saída cada grupo de informações que possua:
  - Um formato diferente;
  - Uma lógica de processamento diferente.

Exemplos:

- Fluxos de dados que se dirigem para as entidades externas;
- Relatórios (batch, tela, impressos);
- Fita ou discos contendo transações geradas pela aplicação.
- Consulta Externa (CE) - Assim como uma SE, é um processo elementar que envia dados (ou informações de controle) para fora da aplicação, mas sem realização de nenhum cálculo nem a criação de dados derivados. Seu objetivo é apresentar informação para o usuário, por meio apenas de uma recuperação das informações. Nenhum ALI é mantido durante sua realização, nem o comportamento do sistema é alterado.

De forma similar, envolve a identificação de funções transacionais (EE, SE e CE) e sua classificação de acordo com a complexidade funcional. A definição da complexidade funcional é feita com base no número de arquivos referenciados e dos itens de dados manipulados pela função.

Um arquivo referenciado pode ser um ALI lido ou mantido pela função transacional, ou um AIE lido pela função transacional. Já o número de itens de dados referenciados é calculado considerando apenas os itens de dados efetivamente referenciados pela função transacional em questão.

A definição de complexidade de Entradas Externas é feita com base na Tabela A.2. Como padrão de contagem da Quantidade de itens de dados, deve-se:

- Contar um item para cada campo não-repetido e reconhecido pelo usuário que entra ou sai da aplicação e é requerido para completar a entrada externa;
- Contar um item para a capacidade de enviar para fora da fronteira uma mensagem do sistema para indicar a ocorrência de erro durante o processamento, confirmando que esse processamento foi completo ou verificando que esse processamento deverá continuar.

Como padrão de contagem de Número de Arquivos Referenciados, deve-se:

- Contar um arquivo para cada ALI mantido.
- Contar um arquivo para cada ALI ou AIE durante o processamento da Entrada Externa.
- Contar somente um arquivo para cada ALI que seja lido e mantido.

Complexidade		QUANTIDADE DE ITENS DE DADOS		
		1 a 4	5 a 15	16 ou mais
Nº A R Q.  R E F E R E N C.	0 a 1	Simples	Simples	Média
	2	Simples	Média	Complexa
	3 ou mais	Média	Complexa	Complexa

TABELA A.2 – COMPLEXIDADE DAS ENTRADAS EXTERNAS (EE)

A definição de complexidade de Saídas Externas da função é feita com base na Tabela A.3. Como padrão de contagem da *Quantidade de itens de dados*, deve-se:

- Contar um item para cada campo não-repetido e reconhecido pelo usuário que entra na fronteira da aplicação e é requerido para especificar quando, qual e/ou como o dado será recuperado ou gerado pelo processo elementar;
- Contar um item para cada campo não-repetido e reconhecido pelo usuário que sai da fronteira da aplicação;
- Se um item entrar e sair da fronteira, deve ser contado somente uma vez para o processo elementar;
- Não contar literais como itens;
- Não contar variáveis de página ou rótulos gerados pelo sistema.

Como padrão de contagem do *Número de Arquivos Referenciados*, deve-se:

- Contar um arquivo para cada ALI ou AIE lido durante o processamento do processo elementar.
- Contar um arquivo para cada ALI mantido durante o processamento do processo elementar.
- Contar somente um arquivo para cada ALI lido e mantido durante o processamento do processo elementar.

Complexidade		QUANTIDADE DE ITENS DE DADOS		
		1 a 4	5 a 15	16 ou mais
Nº A R Q.  R E F E R E N C.	0 a 1	Simples	Simples	Média
	2 a 3	Simples	Média	Complexa
	4 ou mais	Média	Complexa	Complexa

TABELA A.3 – COMPLEXIDADE DAS SAÍDAS EXTERNAS (SE)

Consulta Externa é um processo elementar que envia dados para fora da fronteira da aplicação. A intenção primária de uma CE é apresentar informações ao usuário através da recuperação de dados de um ALI ou AIE. O processamento lógico não contém nenhuma fórmula matemática ou cálculo, ou cria dados derivados e o comportamento do sistema não é alterado. Identifica-se como uma função de consulta, um par de transações que apresentem:

- Uma entrada seguida de uma saída;
- Nenhuma atualização de arquivos;
- Formatos diferentes em relação a outras transações;
- Uma lógica de processamento diferente.

Exemplos:

- Tela de Help;
- Relatórios emitidos em resposta a uma solicitação do usuário.

Como padrão de contagem da *Quantidade de itens de dados*, deve-se:

- Contar um item para cada campo não-repetido que entra na aplicação e é requerido para especificar quando, qual e/ou como o dado será recuperado ou gerado pelo processo elementar;
- Contar um item para cada campo não-repetido que sai da aplicação;
- Se um item entrar e sair da fronteira, conte somente uma vez;
- Não contar literais como itens;

- Não contar variáveis de página ou rótulos gerados pelo sistema.

Como padrão de contagem do *Número de Arquivos Referenciados*, deve-se:

- Contar um item para cada ALI lido durante o processamento elementar;
- Contar um item para cada AIE lido durante o processamento elementar.

A definição de complexidade de Consultas Externas é determinada observando-se separadamente as entradas e saídas. A parte da Consulta referente às entradas é feita a partir da Tabela A.4, já a parte referente às saídas é feita a partir da Tabela A.5.

Complexidade		QUANTIDADE DE ITENS DE DADOS		
		1 a 4	5 a 15	16 ou mais
Nº A R Q.  R E F E R E N C.	1	Simples	Simples	Média
	2 a 3	Simples	Média	Complexa
	4 ou mais	Média	Complexa	Complexa

TABELA A.4 – COMPLEXIDADE DAS CONSULTAS EXTERNAS (CE) - ENTRADAS

Complexidade		QUANTIDADE DE ITENS DE DADOS		
		1 a 5	6 a 19	20 ou mais
Nº A R Q.  R E F E R E N C.	1	Simples	Simples	Média
	2 a 3	Simples	Média	Complexa
	4 ou mais	Média	Complexa	Complexa

TABELA A.5 – COMPLEXIDADE DAS CONSULTAS EXTERNAS (CE) – SAÍDAS

Após a identificação das quantidades em cada uma das tabelas apresentadas, multiplica-se a quantidade de cada tipo de complexidade pelo fator correspondente à Tabela A.6, considerando para as Consultas Externas separadamente as Entradas e Saídas, utilizando apenas o maior resultado entre elas.

Fatores para Cálculo		Complexidade Funcional		
		Simples	Média	Complexa
D E S C R I Ç Ã O	Arquivo Lógico Interno	x 7	x 10	x 15
	Arquivo de Interface Externa	x 5	x 7	x 10
	Entrada Externa	x 3	x 4	x 6
	Saída Externa	x 4	x 5	x 7
	Consulta Externa	x 3	x 4	x 6

TABELA A.6 – FATORES PARA CÁLCULO DE PONTOS DE FUNÇÃO BRUTOS

## 2 – DETERMINAR O VALOR DO FATOR DE AJUSTE

O fator de ajuste é baseado em 14 características gerais de sistemas, que avaliam a funcionalidade geral da aplicação que está sendo contada, e seus níveis de influência, descritas a seguir.

Cada uma destas características pode ter um grau de influência de 0 a 5, dependendo da necessidade e da exigência da característica para o sistema:

0 - Nenhuma influência

1 - Influência mínima

2 - Influência moderada

3 - Influência média

4 - Influência significativa

5 - Influência forte

### 2.1 – Comunicação de Dados

As informações processadas pelo sistema são enviadas e recebidas através de comunicação de dados, mesmo que local. Esta característica se aplica para sistemas que serão executados em redes de teleprocessamento e pode ter os seguintes níveis:

- 0 - Aplicação 100% batch ou micro stand-alone;
- 1 - Aplicação batch mas com impressão remota ou entrada de dados remota;
- 2 - Aplicação batch mas utiliza impressão remota e entrada de dados remota;
- 3 - Entrada de dados on-line para alimentar processamento batch ou consulta;
- 4 - Entrada de dados on-line mas suporta só um tipo de protocolo de comunicação;
- 5 - Entrada de dados on-line e suporta vários tipos de protocolo de comunicação.

## **2.2 – Funções Distribuídas**

Distribuição de processamento e/ou dados em mais de um processador:

- 0 - Não participa de transferência de dados/processamento entre os processadores;
- 1 - Prepara dados, tais como planilhas para o usuário final usar em outro processador;
- 2 - Prepara dados e transfere-os para processamento em outro equipamento;
- 3 - Processamento distribuído e transferência de dados on-line em uma direção;
- 4 - Processamento distribuído e transferência de dados on-line em ambas as direções;
- 5 - O processamento é dinamicamente direcionado para a CPU mais apropriada.

## **2.3 – Desempenho da Execução do Sistema**

Requisitos de desempenho estabelecidos pelo usuário como aceitáveis para o tempo de resposta do sistema:

- 0 - Nenhum requisito especial de desempenho foi solicitado pelo usuário;
- 1 - Requisito de desempenho foi citado, mas sem requerer uma ação especial;
- 2 - Tempo de resposta e volume de processamento são críticos em horários de pico. Sem determinação especial. A data limite do processamento é o próximo dia útil;
- 3 - Tempo de resposta e volume de processamento são críticos no horário comercial. Sem determinação especial. A comunicação com outros sistemas é limitante;
- 4 - Os requisitos de desempenho estabelecidos pelo usuário são rigorosos o bastante para requerer tarefas de análise e projeto da aplicação;
- 5 - Além do descrito no item 4, é necessário o uso de ferramentas de análise de desempenho para garantir a obtenção dos requisitos.

## **2.4 – Carga de Máquina (Utilização do equipamento)**

Característica que mostra a necessidade de se fazer considerações especiais no projeto para que a configuração do equipamento não fique sobrecarregada.

- 0 - Não foi estabelecida nenhuma restrição operacional;
- 1 - Existem restrições operacionais leves, sem esforço extra para suportá-las;
- 2 - Algumas considerações de ajuste de desempenho e segurança são necessárias;
- 3 - Necessidades especiais de processador para uma parte específica do sistema;
- 4 - Restrições operacionais requerem cuidados especiais no processador;
- 5 - Além das características do item 4, há considerações especiais na distribuição do sistema e em seus componentes.

## **2.5 – Volume de Transações**

Avaliação do nível de influência do volume de transações elevado, influenciando o desenvolvimento, instalação e suporte da aplicação.

- 0 - Nenhum período de pico de transações é esperado;
- 1 - Picos de transações mensais, trimestrais, anuais ou sazonais;
- 2 - São previstos picos semanais;
- 3 - São previstos picos diários;
- 4 - Alto volume de transações foi estabelecido pelo usuário, ou o tempo de resposta necessário atinge nível alto o suficiente para requerer análise de desempenho;
- 5 - Além do descrito no item 4, é necessário uso de ferramentas de análise de desempenho para atender aos altos volumes de transação.

## **2.6 – Entrada de Dados On-Line**

Quantificar o nível de influência exercida pela utilização de entrada de dados on-line.

- 0 - Todas as transações são processadas em modo batch;
- 1 - De 1% a 7% das transações são entradas de dados on-line;
- 2 - De 8% a 15% das transações são entradas de dados on-line;
- 3 - De 16% a 23% das transações são entradas de dados on-line;
- 4 - De 24% a 30% das transações são entradas de dados on-line;
- 5 - Mais de 30% das transações são entradas de dados on-line.



## **2.7 – Eficiência do Usuário Final (Interface com o usuário)**

Quantificar o grau de influência relativo aos recursos usados, visando tornar o sistema mais amigável, permitindo melhoria na eficiência do usuário final, tais como:

- Auxílio à navegação (teclas de função, acesso direto e menus dinâmicos);
- Menus, Menus pop-up windows;
- Documentação e help on-line;
- Movimento automático do cursor, movimento de tela vertical e horizontal;
- Impressão remota através de transações on-line;
- Processos batch submetidos a partir de transações on-line;
- Teclas de função pré-definidas, seleção de cursor em campos da tela;
- Uso intenso de vídeo-reverso, intensificados, coloridos e outros efeitos;
- Impressão da documentação das transações on-line através de hard-copy;
- Utilização de mouse ou outros dispositivos apontadores;
- O menor número possível de telas para executar as funções do negócio;
- Suporte bilingue (contar como 4 itens);
- Suporte multilingue (contar como 6 itens).

Classificar o nível de influência conforme abaixo:

0 - nenhum dos itens descritos;

1 - De um a três itens descritos;

2 - De quatro a cinco itens descritos;

3 - Mais de cinco itens, mas sem requisitos específicos relacionados à amigabilidade;

4 - Mais de cinco itens e foram estabelecidos requisitos quanto à amigabilidade. O projeto deve incluir fatores para minimizar a digitação;

5 - Mais de cinco itens e foram estabelecidos requisitos quanto à amigabilidade. O projeto deve incluir o uso de ferramentas e processos especiais para demonstrar que os objetivos de eficiência foram alcançados.

## **2.8 – Atualizações On-Line**

Os Arquivos Lógicos Internos deverão ser atualizados on-line:

0 - Nenhum arquivo atualizado on-line;

1 - Atualização on-line de um a três ALI's. O volume de atualização é baixo e a recuperação de dados é simples;

- 2 – Idem ao item 1 para mais de três ALI's;
- 3 - Atualização on-line da maioria dos ALI's;
- 4 - Idem ao item 3, mais proteção contra perda de dados no sistema;
- 5 - Idem ao item 4, mais recuperação de alto custo devido a altos volumes de dados.

## **2.9 – Processamento Complexo**

Deve ser quantificado o grau de influência da complexidade de processamento, com base nas seguintes categorias:

- Processamento especial de auditoria e/ou de segurança;
- Processamento lógico extensivo;
- Processamento matemático extensivo;
- Processamento gerando muitas exceções, resultando em transações incompletas, que devem ser reprocessadas;
- Processamento complexo para manusear múltiplas possibilidades de entrada/saída.

- 0 – Nenhum dos itens descritos;
- 1 – Apenas um dos itens descritos;
- 2 – Dois dos itens descritos;
- 3 – Três dos itens descritos;
- 4 – Quatro dos itens descritos;
- 5 – Todos os cinco itens descritos.

## **2.10 – Reusabilidade (reutilização de código)**

O sistema e o código gerado serão projetados, desenvolvidos e suportados para serem reutilizados em outros sistemas:

- 0 - Nenhuma preocupação com reutilização de código;
- 1 - Código reutilizado é utilizado somente dentro do próprio sistema;
- 2 - Menos de 10% da aplicação foi projetada prevendo reutilização do código;
- 3 - 10% ou mais da aplicação é projetada prevendo reutilização do código;
- 4 - Aplicação projetada com código reutilizável e o sistema é customizado pelo usuário em nível de código-fonte;

5 - Aplicação projetada para ter seu código facilmente reutilizado por outra aplicação sendo customizada através de parâmetros que podem ser alterados pelo usuário.

### **2.11 – Facilidade de Instalação (implantação)**

Inclusão no projeto de facilidades especiais para agilizar a conversão e implantação.

0 - Nenhuma consideração ou procedimento especial foi estabelecido pelo usuário

1 - Nenhuma consideração especial foi estabelecida pelo usuário, mas procedimentos especiais são necessários na implantação;

2 - Requisitos de conversão e implantação foram estabelecidos pelo usuário e seus roteiros foram providos e testados. O impacto não foi considerado importante;

3 – Idem ao item 2, tendo o impacto da conversão sido considerado importante;

4 - Além do item 2, conversão automática e ferramentas de implantação foram providas e testadas;

5 - Além do item 3, conversão automática e ferramentas de implantação foram providas e testadas.

### **2.12 – Facilidade Operacional**

Eliminação e/ou facilitação de atividades manuais como manuseio de papel, fitas e automação de procedimentos de inicialização, backup e recuperação.

0 – Nenhuma consideração especial com a operação do sistema;

1-4 – Verificar a quantidade de itens que podem ser identificadas na aplicação;

- Necessários procedimentos eficientes de inicialização, backup e recuperação, mas a intervenção do operador é necessária;
- Procedimentos eficientes de inicialização, backup e recuperação são necessários sem nenhuma intervenção do operador (contar como dois itens);
- A aplicação minimiza a operação de montagem de fitas magnéticas;
- A aplicação minimiza a necessidade de manuseio de papel.

5 - O sistema está sendo projetado para não precisar da intervenção do operador no seu funcionamento normal. Apenas a inicialização e parada do sistema ficam a cargo do operador. A recuperação automática de erros é uma característica do sistema.

### **2.13 – Operação do Sistema em Múltiplos Locais**

O sistema será projetado visando sua operação em múltiplos locais.

0 - Nenhum requisito do usuário;

1 - Necessidade de instalação em múltiplos locais foi considerada no projeto e o sistema está preparado para operar sobre o mesmo ambiente de software e hardware;

2 - Foi considerada a necessidade de instalação em múltiplos locais e o sistema está preparado para operar apenas em ambientes similares de software e hardware;

3 - Foi considerada a necessidade de instalação em múltiplos locais e o sistema está preparado para operar sob diferentes ambientes de software e hardware;

4 - Plano de documentação e manutenção foram providos e testados para suportar o sistema em múltiplos locais. Além disso, os itens 1 ou 2 caracterizam a aplicação;

5 - Plano de documentação e manutenção foram providos e testados para suportar o sistema em múltiplos locais. Além disso, o item 3 caracteriza a aplicação;

### **2.14 – Facilidade de Mudanças (Flexibilidade)**

Uso de dispositivos para facilitar a alteração e manutenção do sistema. As seguintes características podem ser atribuídas à aplicação:

- Facilidades como consultas e relatórios flexíveis para atender necessidades simples (contar como 1 item);
- Facilidades como consulta e relatórios flexíveis para atender necessidades de complexidade média (contar como 2 itens);
- Facilidades como consulta e relatórios flexíveis para atender necessidades de complexas (contar como 3 itens);
- Dados de controle são armazenados em tabelas que são mantidas pelo usuário através de processos on-line, mas as mudanças são efetivas no dia seguinte;
- Idem ao item anterior com efeito imediato (contar como 2 itens).

0 - Nenhum dos itens descritos;

1 - Um dos itens descritos;

2 - Dois dos itens descritos;

3 - Três dos itens descritos;

4 - Quatro dos itens descritos;

5 - Todos os itens.

### 3 – CÁLCULO DOS PONTOS DE FUNÇÃO AJUSTADOS

Os 14 graus de influência informados são somados, resultando no nível de influência total e o Valor do Fator de Ajuste é determinado então pela equação apresentada na Fórmula A1.

$\text{VFA} = \left( \sum_{i=1}^{14} \text{GI}_i * 0,01 \right) + 0,65$
<p>onde: VFA = Valor do Fator de Ajuste; I = Grau de Influência de cada característica geral de sistema.</p>

FÓRMULA A.1 - VALOR DO FATOR DE AJUSTE DA APF

Como a somatória dos graus de influência resultará em um valor entre 0 e 70, o Fator de Ajuste visa ajustar o PFNA em  $\pm 35\%$ .

## ANEXO B

### DETALHAMENTO DA MÉTRICA PONTOS POR USE CASE (UCP)

Conforme Ribu (2001), os UCP podem ser obtidos a partir da análise dos casos de uso que compõem o sistema. O Processo de Contagem de UCP é composto de seis etapas.

#### 1 - TOTAL DE PESOS NÃO AJUSTADOS DOS ATORES

Deve-se classificar os atores de acordo com a Tabela B.1, multiplicando a quantidade de atores de cada tipo, pelo fator correspondente, obtendo o UAW (*Unadjusted Actor Weights*), com a somatória desses produtos.

Tipo de Ator	Descrição	Fator
SIMPLES	Sistemas Externos	1
MÉDIO	Hardware ou temporizadores	2
COMPLEXO	Humanos	3

TABELA B.1 – CLASSIFICAÇÃO DOS ATORES. Fonte Ribu (2001)

#### 2 - TOTAL DE PESOS NÃO AJUSTADOS DOS CASOS DE USO

Cada Caso de Uso é classificado de acordo com a Tabela B.2, dependendo de sua quantidade de transações, incluindo-se cenários secundários, multiplicando a quantidade de Casos de Uso de cada tipo, pelo fator correspondente, obtendo o UUCW (*Unadjusted Use Case Weights*), com a somatória desses produtos.

Tipo de Caso de Uso	Descrição	Fator
SIMPLES	Até 3 caminhos	5
MÉDIO	De 4 a 7 caminhos	10
COMPLEXO	Acima de 7 caminhos	15

TABELA B.2 – CLASSIFICAÇÃO DOS CASOS DE USO. Fonte Ribu (2001)

### 3 - PONTOS TOTAIS NÃO AJUSTADOS

Calcula-se então o Total de Pontos Não Ajustados, através da soma apresentada na Fórmula B.1:

<b><math display="block">UUCP = UAW + UUCW</math></b>
onde : UUCP = Unadjusted Use Case Points (Total de Pontos Não Ajustados); UAW = Unadjusted Actor Weights (Pesos Não Ajustados dos Atores); UUCW = Unadjusted Use Case Weights (Pesos Não Ajustados dos Use Cases).

FÓRMULA B.1 – TOTAL DE PONTOS NÃO AJUSTADOS DA UCP

### 4 - DETERMINAR O FATOR DE COMPLEXIDADE TÉCNICA

São considerados os fatores técnicos, descritos na Tabela B.3, atribuindo-lhes uma avaliação de 0 a 5 para cada fator. A avaliação 0 indica que o fator é irrelevante, enquanto a avaliação 5 indica que o mesmo é essencial.

Fator Técnico	Peso
Sistema Distribuído	2
Tempo de Resposta ( desempenho)	1
Eficiência do Usuário Final	1
Complexidade do processamento interno	1
Código reutilizável	1
Fácil instalação	0.5
Fácil de usar	0.5
Portátil	2
Fácil de mudar	1
Concorrente	1
Características especiais de segurança	1
Acesso direto ao software	1
Necessidade de treinamento especial para o usuário	1

TABELA B.3 – AVALIAÇÃO DOS FATORES TÉCNICOS. Fonte Ribu (2001)

Após a avaliação de cada tópico, multiplica-se a avaliação efetuada pelo peso correspondente, obtendo-se o Total de Fator Técnico. O Fator de Complexidade Técnica é obtido através da Fórmula B2.

$TCF = (0,6 + (0,01 * \sum TF))$
<p>onde: TCF = <i>Technical Complexity Factor</i> (Fator de Complexidade Técnica);  TF = <i>Technical Factor</i> (Fator Técnico).</p>

FÓRMULA B.2 - FATOR DE COMPLEXIDADE TÉCNICA DA UCP

## 5 - DETERMINAR A EFICIÊNCIA DO FATOR AMBIENTAL

Esse fator considera o nível de experiência dos membros da equipe do projeto e tem os pesos conforme a Tabela B.4, devendo-se também atribuir uma avaliação de 0 a 5 para cada fator.

Fator Ambiental	Peso
Usando um processo formal de desenvolvimento (metodologia)	1,5
Usuários têm experiência com algum aplicativo anterior	0,5
Experiência orientada a objeto	1
Capacidade do Analista Chefe	0,5
Motivação	1
Requisitos Estáveis	2
Trabalhadores em tempo parcial	-1
Dificuldade de linguagem de programação	-1

TABELA B.4 – AVALIAÇÃO DOS FATORES AMBIENTAIS. Fonte Ribu (2001)

O Fator de Complexidade Ambiental é obtido através da equação apresentada na Fórmula B.3:

$ECF = (1,4 + (-0,03 * \sum EF))$
<p>onde: ECF = <i>Environment Complexity Factor</i> (Fator de Complexidade Ambiental);  EF = <i>Environment Factor</i> (Fator Ambiental);</p>

FÓRMULA B.3 - FATOR DE COMPLEXIDADE AMBIENTAL



## 6 - PONTOS TOTAIS DE CASOS DE USO AJUSTADOS

O cálculo dos Pontos de Caso de Uso é efetuado a partir dos três componentes já calculados, conforme apresentado na Fórmula B4.

$$UCP = UUCP * TCF * ECF$$

onde: UCP = *Use Case Points* (Total de Pontos por Caso de Uso);  
UUCP = *Unadjusted Use Case Points* (Total de Pontos Não Ajustados);  
TCF = *Technical Complexity Factor* (Fator de Complexidade Técnica);  
ECF = *Environment Complexity Factor* (Fator de Complexidade Ambiental).

FÓRMULA B.4 – PONTOS TOTAIS DE CASO DE USO AJUSTADOS DA UCP

## **7 - REFERÊNCIAS BIBLIOGRÁFICAS:**

[Abran (1999)] ABRAN, A. Full function point measurement manual: V2.0. Canadá: Software Engineering Laboratory in Applied Metrics - Universidade de Quebec, 1999.

[Abran (2003)] ABRAN, A., et al. 'COSMIC-FFP Measurement Manual v2.2, The COSMIC Implementation Guide for ISO/IEC 19761:2003'

[Aguiar (2002)] AGUIAR,M. Estimando os projetos com COCOMO II. Developers Magazine – Set,2002.

[Aguiar (2003)] AGUIAR,M. Pontos de Função ou Pontos por Caso de Uso? Como Estimar Projetos Orientados a Objetos. Developers Magazine – Jun,2003.

[August (1993)] AUGUST,J.H. JAD – Joint Application Design. São Paulo, Makron Books,1993.

[Boehm (1981)] BOEHM,B. – Software Engineering Economics – Prentice-Hall, 1981.

[Boehm (2000)] BOEHM,B. et al. – Software Cost Estimation With COCOMO II” – Prentice Hall, 2000.

[Calazans (2005)] CALAZANS,A. – Avaliação das Características Gerais de Sistemas na Análise por Pontos de Função – APF por meio da aplicação do GQM Goal, Questions, Metrics, VII Simpósio Internacional de Melhoria de Software, São Paulo, Novembro de 2005.

[Carvalho (2001)] CARVALHO, J.G., Praxis Mentor - Uma Ferramenta de Apoio à Utilização de um Processo de Desenvolvimento de Software. Dissertação de Mestrado, UFMG - Departamento de Ciência da Computação. Belo Horizonte, 2001

[CHAOS (2003)] THE STANDISH GROUP. CHAOS Report Shows Project Success Rates Have Improved by 50%. Disponível em <http://www.standishgroup.com/press/article.php?id=2> Acesso em 20.07.2005.

[Chinelato (2004)] CHINELATO FILHO,J. – O&M Integrado à Informática. Rio de Janeiro: LTC Livros Técnicos e Científicos, 2004.

[Cordeiro (2002)] CORDEIRO, M.A., Uma Ferramenta Automatizada de suporte ao processo de Gerenciamento de Requisitos. Dissertação de Mestrado, PUCPR. Centro de Ciências Exatas e de Tecnologia, CCET. Curitiba, 2002

[Costa (1994)] COSTA,O.W.D. JAD - Joint Application Design. Rio de Janeiro, Livraria e Editora Infobook, 1994.

[Dekkers (2003)] DEKKERS, C. A. Measuring the “logical” or “functional” Size of Software Projects and Software Application. Spotlight Software, ISO Bulletin – Maio de 2003.

[Dias (2003)] DIAS,R. Análise por Pontos de Função: Uma Técnica para Dimensionamento de Sistemas de Informação on-line. Disponível em: <http://www.presidentekennedy.br/resi/edicao03/artigo02.pdf>. Acessado em 31.08.2005.

[Falbo (2005)] FALBO,R.A. Análise de Pontos de Função – Notas de Aula. Disponível em [www.inf.ufes.br/~falbo/download/aulas/es-g/2005-1/APF.pdf](http://www.inf.ufes.br/~falbo/download/aulas/es-g/2005-1/APF.pdf). Acessado em 31.08.2005.

[Fernandes (1992)] FERNANDES,A.A. e ALVES,M.M. Gerência Estratégica da Tecnologia da Informação. Rio de Janeiro: LTC Livros Técnicos e Científicos, 1992.

[Fernandes (1995)] FERNANDES,A.A.; Gerência de Software Através de Métricas; São Paulo: Atlas, 1995.

[Fiorini (1998)] FIORINI,S.T., et al. Anais do WER98 - Workshop em Engenharia de Requisitos, Maringá-PR, Brasil, 1998

[Gomoll (1990)] GOMOLL, K. *Some techniques for observing users*. In "The art for human-computer interface design" Ed. Addison-Wesley Publishing Company. Massachusetts, 1990.

[Hazan (2001)] HAZAN.C.; et al. Medição da Qualidade e Produtividade em Software, In: Qualidade e Produtividade em Software, 4ª ed., Makron Books, 2001

[IEEE (1983)] IEEE Standard Glossary of Software Engineering Terminology, Nova Iorque: IEEE, ANSI/IEEE Std 729.1983, 1983.

[IFPUG (2005)] IFPUG - International Function Point Users Group. Function Point Counting Practices Manual (CPM), Release 4.2.1. Janeiro, 2005.

[ISO (2003)] ISO/IEC 20926:2003 - Software engineering -- IFPUG 4.1 Unadjusted functional size measurement method -- Counting practices manual

[Jacobson (1992)] JACOBSON,I. , et al. Object-Oriented Software Engineering: A Use Case Driven Approach - Ed. Pearson, 1994

[Jurison (1999)] JURISON,J. Software Project Management: The Manager's View. Communications of AIS (Volume 2, Article 17), 1999.

[Kemerer (1987)] KEMERER,C.F. An Empirical Validation of Software Cost Estimation Models. Communications of the ACM, v.30 n.5, Maio de 1987

[Kotonya (1998)] KOTONYA, G. and I. Sommerville. Requirements Engineering: Processes and Techniques. New York, 1999.

[Larman (2004)] LARMAN,C. Utilizando UML e Padrões 2ª Edição – Porto Alegre-Ed. Bookman, 2004.

[McGarry (2002)] McGARRY,J. et al. – Practical Software Measurement: Objective Information for Decision Makers. Ed. AddisonWesley Longman, 2002.

[NESMA (2005)] Netherlands Software Metrics Association – Disponível em: <http://www.nesma.nl> Acessado em 31.08.2005.

[Page-Jones (2001)] PAGE-JONES,M. Fundamentos do desenho orientado a objeto com UML, Makron Books, 2001

[Paula Filho (2003)] PAULA FILHO,W.P. Engenharia de Software: Fundamentos, Métodos e Padrões, 2ª ed., Rio de Janeiro: LTC Livros Técnicos e Científicos, 2003.

[PMBok (2004)] PMBoK - Um Guia do Conjunto de Conhecimentos em Gerenciamento de Projetos – Terceira Edição, Project Management Institute, 2004

[PMI (2006)] Project Management Institute – Disponível em: [www.pmi.org](http://www.pmi.org) Acessado em 30.04.2006.

[PMIBR (2006)] PMI no Brasil – Disponível em: [www.pmi.org.br](http://www.pmi.org.br) Acessado em 30.04.2006.

[PMISP (2006)] Project Management Institute – Chapter SP – Disponível em: [www.pmispp.org.br](http://www.pmispp.org.br) Acessado em 30.04.2006.

[Pressman (2006)] PRESSMAN, R.S. Engenharia de Software 6ª edição. São Paulo, McGraw Hill, 2006.

[Reifer (1998)] REIFER,D.; BOEHM,B.; CHULANI,S.; The Rosetta Stone: Making COCOMO 81 Files Work With COCOMO II; EUA: Centro de pesquisa em engenharia de software da Universidade do Sul da Califórnia, 1998.

[Ribu (2001)] RIBU, K. Estimating Object-Oriented Software Projects with Use Cases. Dissertação de Mestrado, Departamento de Informática da Universidade de Oslo. Oslo, 2001.

[Roetzheim (1998)] ROETZHEIM,W.H. e REYNA,A.B. – Software Project Cost & Schedule Estimating – Prentice-Hall, 1998.

[SAP (2006)] SAP (Systemanalyse and Programmentwicklung) – Disponível em: [www.sap.com](http://www.sap.com). Acessado em 30.04.2006.

[SEI (2006)] SEI - Software Engineering Institute - Disponível em: [www.sei.cmu.edu/cmml](http://www.sei.cmu.edu/cmml). Acessado em 28.05.2006.

[Sommerville (1997)] SOMMERVILLE I. – Requirements Engineering – A good practice guide, 1997.

[USC (2005)] USC Center for Software Engineering. The University of Southern California. Disponível em <http://sunset.usc.edu/cse/>. Acessado em 20/09/2005.

[Vavassori (2002)] VAVASSORI,F.B. Metodologia para o Gerenciamento Distribuído de Projetos e Métrica de Software. Tese de Doutorado em Engenharia de Produção – UFSC, Florianópolis, SC, 2002.

[Wilson (1998)] WILSON,M.G. Gestion de Riesgos y sus Aplicaciones, los Metodos S:PRIME y S:PLAN. Anais da IX Conferência Internacional de Tecnologia de Software – Qualidade de Software. Curitiba, pgs. 99-116, 1998.

[Xavier (2005)] XAVIER, C.M. et al. – Metodologia de Gerenciamento de Projetos – METHODWARE. Rio de Janeiro: Brasport, 2005.

[Yourdon (1990)] YOURDON, E. Análise Estruturada Moderna. Rio de Janeiro: Campus, 1990.

## APÊNDICE A – MODELO DE PESQUISA DE PROJETOS CONCLUÍDOS

**NUM.PROJETO:**

**TÍTULO:**

**ANALISTA:**

**LÍDER:**

**PLANEJADO - INÍCIO:**                      **TÉRMINO:**                      **DURAÇÃO:**    **dias**

**REALIZADO - INÍCIO:**                      **TÉRMINO:**                      **DURAÇÃO:**    **dias**

**ÁREA CLIENTE:**                      **ATRASSO:**                      **% ERRO:**

**TIPO DE DESENVOLVIMENTO:**

	0 %	10 a 50%	60 a 90%	100%	Não Lembro
Quanto ao % de utilização da Metodologia de Desenvolvimento Sistemas...					
Quanto à capacitação da equipe do projeto, na utilização das ferramentas e técnicas recomendadas pela Metodologia...					
Quanto às interferências do pessoal de informática em outras atividades não relacionadas ao projeto, em relação ao tempo realizado...					
Quanto ao envolvimento dos usuários durante o desenvolvimento do projeto...					

<b>COM RELAÇÃO A ALTERAÇÕES NOS REQUISITOS DEFINIDOS NA FASE DE ANÁLISE (PROJETO LÓGICO):</b>	<b>SIM</b>	<b>NÃO</b>	<b>Não Lembro</b>
O sistema correspondeu exatamente aos Requisitos Originais indicados pelos usuários na etapa de Levantamento de Requisitos.			
Após a Implementação, foram identificados requisitos que haviam sido solicitados na etapa de Levantamento de Requisitos e que não foram contemplados na versão original do sistema.			
Após a Implementação, os usuários alegaram que faltavam requisitos que mesmo não tendo sido discutidos na fase de Análise, entendiam que seriam considerados na versão original do sistema.			
Após a Implementação, alguns usuários ficaram frustrados com a não inclusão na versão original do sistema, de requisitos que foram discutidos após a fase de Análise, sendo considerado pela equipe desenvolvedora como solicitação de manutenção posterior.			
Alterações nos Requisitos em funções já testadas provocaram erros que apareceram depois da implantação.			
Observação significativa:			



<b>VOCÊ CONSIDERA QUE O SISTEMA APÓS A IMPLANTAÇÃO:</b>	<b>SIM</b>	<b>NÃO</b>	<b>Em Parte</b>
Atendeu às necessidades da área cliente.			
Excedeu demasiadamente os custos previstos inicialmente.			
Gerou um lucro significativo para a empresa.			
Gerou um ganho significativo no processo da área cliente.			
Apresentou excesso de falhas nas funções implementadas.			
Apresentou a necessidade de novas funcionalidades, até então não identificadas.			
Observação significativa:			

<b>QUANTO AO ATRASO OCORRIDO, AS PRINCIPAIS CAUSAS FORAM:</b>	<b>SIM</b>	<b>NÃO</b>	<b>Não Lembro</b>
Alterações de Requisitos por parte dos usuários.			
Atraso em outro projeto com restrição de implantação anterior ou simultânea.			
Falta de recursos pessoais, disponíveis para o projeto.			
Atraso na aquisição de hardware, software necessários.			
Falta de disponibilidade dos usuários, durante o desenvolvimento, treinamento ou implantação.			
Excesso de erros encontrados nas fases de testes e treinamento.			
Falta de verba necessária para cobrir custos imprevistos.			
Falha de comunicação entre a equipe de desenvolvimento e as partes envolvidas.			
Outra causa não citada:			

## APÊNDICE B – ATA DE REUNIÃO JAD

DATA: 09 e 10/05/2005

HORÁRIO: 09h00 às 12h00

LOCAL: Cubatão – Prédio Administrativo – 3º andar – Sala de reuniões da área de T.I.

ASSUNTO: Anteprojeto – Apoio à Gestão de Escopo de Projeto

Participantes:	NOME	FUNÇÃO
	Antonio Carlos Guerra	Analista de Sistemas
	Sandro Gomes Bartolomeu	Analista de Sistemas
	Edilson Martins Rodrigues	Analista de Sistemas
	José Maurício Fernandes	Analista de Sistemas
	Luciana Aparecida Nogueira	Analista de Sistemas
	Marcos Antônio Santos Simões	Programador
	Jorge Luiz da Silva Neto	Usuário de Sistemas
	Luis Carlos Nogueira Sobrinho	Usuário de Sistemas
	Regina Lúcia Cavalcanti	Documentadora do JAD
	Sergio Figueiredo Pereira	Condutor do JAD

### PAUTA:

1. Abertura dos trabalhos
2. Levantamento da Situação atual na área de Desenvolvimento de Sistemas
3. Definição dos Objetivos
4. Identificação de dificuldades
5. Definição dos Requisitos do Projeto
6. Definição dos Próximos Passos

### 1. Abertura dos Trabalhos

Agradecendo a presença de todos, o Guerra assumiu a abertura da reunião reforçando os objetivos, conforme conversado com cada um na oportunidade do convite realizado. Mostrou que as características desse trabalho são atípicas, não existindo a figura do Executivo Patrocinador, pois o aplicativo a ser desenvolvido será de uso próprio, além de servir de experimento em seus estudos.

Ressaltou a importância da experiência de cada um dos presentes no desenvolvimento de projetos de sistemas, tanto da parte dos profissionais de T.I. quanto como usuários que já vivenciaram as razões que geram as necessidades de mudanças e as consequências da falta de controle do escopo, tanto no tempo de desenvolvimento quanto na qualidade do produto final.

Solicitou ao Figueiredo que assumisse a condução da reunião, apresentando-me também como documentadora da reunião.

Foi consensado pelo condutor com os presentes, as datas e horários das próximas sessões, obtendo o compromisso da disponibilidade de todos.

## **2. Levantamento da Situação atual na área de Desenvolvimento de Sistemas**

Iniciado um brainstorm sobre as participações de cada um em projetos anteriores, comentando-se o que acontece normalmente no desenvolvimento de projetos. Ressaltado nesse tópico:

- a grande quantidade de mudanças no escopo e em funções já definidas, gerando retrabalho, perda de tempo e conseqüente aumento no custo do desenvolvimento. Como causas foi destacada a falha na fase do levantamento de requisitos, porém em muitos casos acontecem por mudanças operacionais e estratégicas no negócio e em alguns casos, mudanças na legislação;
- a falta de formalização das alterações, faz com que algumas vezes nem o próprio líder do projeto tenha certeza se a solicitação de um stakeholder está valendo, além de idas e voltas nas definições por causa de uma falta de análise mais aprofundada do impacto da solicitação;
- O analista Edilson destacou também as cobranças que tem sofrido por não cumprimento de prazos, sem que seja levado em consideração, que o escopo implantado em alguns casos foi dobrado ou triplicado pelos próprios usuários no decorrer do projeto.
- Sobrinho rebateu alegando que independentemente de mudanças, nunca havia visto um projeto que cumpriu os prazos estipulados.

## **3. Definição dos Objetivos**

- Auxiliar na estimativa de prazos e custo do projeto (principal);
- Manter controle total do escopo do projeto, seus requisitos e mudanças aprovadas;
- Apresentar de forma on-line o escopo definido e a situação atual do projeto.

## **4. Identificação de dificuldades**

Por orientação do condutor, passou-se ao levantamento das dificuldades encontradas no desenvolvimento de projetos, sendo relacionados os seguintes tópicos:

- Falta de envolvimento de usuários necessários ao projeto;
- Falta de definição clara de quem é stakeholder do projeto;
- Descontrole de solicitações de mudanças, muitas vezes gerando falsas expectativas e decepções no momento da implantação;
- Mudanças que após iniciadas, conclui-se que não devem ser realizadas, gerando trabalho e retrabalho imprecidentes;
- Falta de um critério unificado para definição de duração dos projetos, gerando erros freqüentemente;
- Falta de conhecimento histórico da produtividade dos desenvolvedores, para facilitar as estimativas dos projetos seguintes;
- Não estimativa e controle de custo do projeto, o que serviria para demonstrar melhor a viabilidade ou não de seu desenvolvimento;
- Falha no posicionamento do andamento dos trabalhos, gerando desmotivação nos usuários envolvidos com o projeto;
- Não comprometimento do Sponsor com as mudanças solicitadas após a formalização do escopo no final do anteprojeto.

## 5. Definição dos Requisitos do Projeto

Foram levantados e posteriormente priorizados os requisitos do projeto:

Manter controle de todos os requisitos levantados e devidamente consensados com os envolvidos na fase de Levantamento de Requisitos, após a devida aprovação do Sponsor;	Essencial
Identificar de forma unívoca os requisitos do projeto, mantendo controle sobre a sua situação, seu tamanho e custo, previstos antes de sua aprovação;	Essencial
Permitir ao Gerente do Projeto o registro da análise técnica da solicitação e se esta apresentar-se inviável tecnicamente, cancelá-la deixando registrado o motivo de tal cancelamento.	Essencial
Permitir a estimativa do tamanho, do tempo e do custo, previstos para a mudança solicitada, para servir de apoio ao Sponsor na análise e decisão de aprovar ou não a Solicitação de mudança no Projeto;	Essencial
Fornecer subsídios de apoio ao Gerente do Projeto, com base nos dados reais de projetos anteriores, quanto às estimativas de esforço no atendimento das solicitações de mudanças;	Essencial
Permitir ao Gerente do Projeto, o registro do impacto das solicitações no projeto como um todo, após sua análise de disponibilidades de recursos e revisão da rede PERT e do Cronograma do Projeto;	Essencial
Permitir ao Sponsor, o registro da aprovação ou recusa da solicitação com base na análise técnica e na análise do impacto, assumindo no caso de aprovação, os novos prazos e orçamento do projeto;	Essencial
Manter todos os dados reais relevantes do projeto armazenados, para servirem de base para futuros projetos;	Essencial
Manter a rastreabilidade dos requisitos do projeto, e controle total do seu escopo;	Essencial
Conhecer os stakeholders do projeto, com o nível de competência dos mesmos, quanto à participação no projeto e permissão de solicitação de mudança e/ou aprovação dos requisitos do projeto;	Importante
Manter controle dos custos realizados no projeto e de forma clara, preferencialmente gráfica, apresentar um comparativo com o custo previsto;	Importante
Permitir aos stakeholders autorizados, o registro diretamente no sistema, de solicitações de mudanças de requisitos definidos;	Desejável
Controlar as mudanças solicitadas e não aprovadas, com as devidas justificativas.	Desejável
Permitir consultas sobre a situação do projeto, acessível a todos os stakeholders;	Desejável

## 6. Definição dos próximos passos

Conforme a metodologia de desenvolvimento de sistemas, será elaborada a documentação técnica, que contará com todos esses tópicos e servirá como base para o desenvolvimento do projeto proposto. Para tal, alguns pontos certamente precisarão ser mais detalhados com alguns dos presentes, o que acontecerá em reuniões específicas a serem marcadas.

**APÊNDICE C**

**SISTEMA DE  
APOIO À GESTÃO DO ESCOPO DO PROJETO  
  
SAGEP**

**DOCUMENTAÇÃO TÉCNICA DO PROJETO**

**Versão 1.0**

## Índice

1 - INTRODUÇÃO .....	3
1.1 - Visão Geral deste documento .....	3
1.2 - Convenções, termos e abreviações .....	3
1.2.1 - Identificação dos Requisitos .....	3
1.2.2 - Prioridade dos Requisitos .....	3
2 - DESCRIÇÃO GERAL DO SISTEMA .....	4
3 - REQUISITOS FUNCIONAIS	
3.1 - Levantamento de Requisitos .....	5
3.2 - Diagrama de Casos de Uso .....	6
3.3 - Detalhamento dos Casos de Uso.....	8
3.3.1 - Cadastrar Projeto.....	8
3.3.2 - Cadastrar Stakeholder.....	9
3.3.3 - Associar Stakeholder ao Projeto.....	10
3.3.4 - Atualizar Custo da Mão de Obra.....	11
3.3.5 - Cadastrar Tipo de Recurso.....	12
3.3.6 - Cadastrar Escopo Inicial.....	13
3.3.7 - Solicitar Mudança.....	14
3.3.8 - Analisar Solicitação.....	15
3.3.9 - Incluir novo Use Case.....	16
3.3.10- Alterar Use Case.....	17
3.3.11- Excluir Use Case.....	18
3.3.12- Estimar impacto de Solicitação.....	19
3.3.13- Avaliar Solicitação.....	20
3.3.14- Apurar dados reais do Use Case.....	21
3.3.15- Avaliar andamento do projeto.....	22
3.3.16- Consultar Use Cases do projeto.....	23
3.3.17- Consultar os dados de um Use Case.....	24
3.3.18- Consultar solicitações de mudanças.....	25
3.3.19- Consultar os dados de uma solicitação.....	26
3.3.20- Consultar UC afetados por uma solicitação.....	27
3.3.21- Consultar histórico de Use Cases similares.....	28
3.3.22- Consultar detalhes de Histórico de Use Cases.....	29
3.3.23- Consultar recursos utilizados em um Use Case.....	30
3.3.24- Consultar situação do projeto.....	31
3.3.25- Calcular Pontos de Função.....	32
3.4 - Diagrama de Transição de Estado.....	33
4 - REQUISITOS NÃO FUNCIONAIS	
4.1 - Usabilidade .....	34
4.2 - Segurança .....	34
4.3 - Disponibilidade .....	34

## 1 - INTRODUÇÃO

Este documento especifica o sistema **APOIO À GESTÃO DO ESCOPO DO PROJETO - SAGEP**, fornecendo aos desenvolvedores as informações necessárias para o projeto e implementação, assim como para a realização dos testes e homologação do sistema

### 1.1 Visão geral deste documento

A introdução fornece informações necessárias para fazer um bom uso deste documento. As demais seções apresentam a especificação do **SAGEP** e estão organizadas como descrito abaixo:

Seção 2 – Descrição geral do sistema: apresenta uma visão geral do sistema, caracterizando qual é o seu escopo e descrevendo seus usuários;

Seção 3 – Requisitos funcionais: especifica os requisitos funcionais do sistema, descrevendo os casos de uso e diagramas do projeto e análise do sistema;

Seção 4 – Requisitos não funcionais: especifica os requisitos não funcionais do sistema, divididos em requisitos de usabilidade, segurança e disponibilidade.

### 1.2 Convenções, termos e abreviações

A correta interpretação deste documento exige o conhecimento de algumas convenções e termos específicos, que são descritos a seguir:

**ESCOPO** – Abrangência do projeto que define o que faz parte desse trabalho;

**GP** – Gerente do Projeto;

**GRID** – Tabela de dados apresentada na tela;

**SPONSOR** – Também denominado Executivo Patrocinador, é o responsável pelas decisões e aprovações relativas ao projeto;

**STAKEHOLDER** – Toda e qualquer pessoa envolvida com o projeto, tanto como cliente ou usuário direto ou indireto do sistema, quanto como desenvolvedor.

### 1.3 Prioridades dos Requisitos

Para estabelecer a prioridade dos requisitos foram adotadas as denominações: “essencial”; “importante” e “desejável”.

**Essencial** é o requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são requisitos imprescindíveis, que têm que ser implementados para que o objetivo principal do projeto seja atingido.

**Importante** é o requisito sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Requisitos importantes devem ser implementados, mas, se não forem, o sistema poderá ser implantado e usado mesmo que precariamente.

**Desejável** é o requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ele. Requisitos desejáveis são requisitos que podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que está sendo especificada.

## **2 - DESCRIÇÃO GERAL DO SISTEMA**

O **SAGEP** é um sistema com a finalidade de auxiliar o Gerente de Projetos a gerenciar as mudanças no escopo do projeto dando suporte às estimativas dos impactos na duração e custos do projeto, com base no histórico de projetos similares anteriores. Não se propõe a eliminar a necessidade de tal profissional especializado nem mesmo assumir funções e decisões de sua alçada, mas servir de ferramenta para auxiliá-lo no desempenho de sua atividade.

Na primeira fase do projeto chamada de Anteprojeto, devem ser levantados os requisitos essenciais do projeto, a fim de se definir o escopo do mesmo e submetê-lo à aprovação do Sponsor.

Após a aceitação do escopo, deverão ser registradas no sistema as funções acordadas, assim como suas previsões de duração e custo. Cada função será identificada de forma unívoca permitindo melhor gerenciamento dos mesmos.

Por motivos diversos, durante o desenvolvimento e até mesmo a implantação do sistema, acontecem solicitações de mudanças, originadas por qualquer stakeholder. Essas mudanças poderão ser a inclusão de novas funções, alteração em funções existentes e até mesmo a exclusão de alguma função.

O próprio solicitante poderá registrar a solicitação no sistema, descrevendo-a e indicando uma justificativa para tal mudança no que foi anteriormente definido. Tal solicitação poderá impactar em uma ou mais aplicações do projeto em desenvolvimento, ficando no aguardo do estudo a ser efetuado pelo Gerente do Projeto.

Uma solicitação ainda não aprovada pelo Sponsor poderá ser cancelada pelo Gerente do Projeto por inviabilidade técnica, ou mesmo por consenso junto ao solicitante. Se considerada viável, o Gerente do Projeto efetua a análise da solicitação e define a necessidade de novas funções no projeto, alterações ou exclusões de Use Cases existentes. Para cada novo Use Case ou Use Case a ser alterado é calculada sua pontuação em Pontos de Função, com o auxílio de uma aplicação desenvolvida para efetuar esses cálculos.

Cabe ao Gerente do Projeto, a partir do tamanho da alteração avaliada e com o auxílio da Consulta à Base Histórica de Use Cases similares, estimar o tempo necessário para o atendimento da solicitação e juntamente com a análise da disponibilidade dos recursos necessários e do Diagrama de Rede do projeto, avaliar possíveis variações na data final do projeto. O resultado dessa análise é lançado pelo Gerente do Projeto no sistema, para servir de suporte à decisão do Sponsor quanto à sua aprovação ou não. Somente se a Solicitação for aprovada pelo Sponsor, o escopo do sistema é alterado, assumindo-se o novo prazo e custo total do projeto.

O Gerente do Projeto deverá também periodicamente registrar sua avaliação quanto ao tempo e custo previstos para o projeto, assim como o percentual desenvolvido. Os dados reais relativos aos Use Cases do projeto deverão ser registrados após sua conclusão, para alimentarem a base de consulta em projetos futuros.

O sistema permitirá além da referida Consulta à base Histórica, Consultas e relatórios de Use Cases de um Projeto, Solicitações de Mudanças e Situação Atual do Projeto.



### **3 - REQUISITOS FUNCIONAIS**

#### **3.1 – LEVANTAMENTO DE REQUISITOS**

##### **ESSENCIAIS:**

- Manter controle de todas as funções levantadas e devidamente consensadas com os envolvidos na fase de Levantamento de Requisitos, após a devida aprovação do Sponsor;
- Identificar de forma unívoca as funções do projeto, mantendo controle sobre a sua situação, seu tamanho e custo, previstos antes de sua aprovação;
- Permitir ao Gerente do Projeto o registro da análise técnica da solicitação e se esta apresentar-se inviável tecnicamente, cancelá-la deixando registrado o motivo de tal cancelamento;
- Permitir a estimativa do tamanho, do tempo e do custo, previstos para a mudança solicitada, para servir de apoio ao Sponsor na análise e decisão de aprovar ou não a Solicitação de mudança no Projeto;
- Fornecer subsídios de apoio ao Gerente do Projeto, com base nos dados reais de projetos anteriores, quanto às estimativas de esforço no atendimento das solicitações de mudanças;
- Permitir ao Gerente do Projeto, o registro do impacto da solicitação no projeto como um todo, após sua análise de disponibilidades de recursos e revisão da rede PERT e do Cronograma do Projeto;
- Permitir ao Sponsor, o registro da aprovação ou recusa da solicitação com base na análise técnica e na análise do impacto, assumindo no caso de aprovação, os novos prazos e orçamento do projeto;
- Manter todos os dados reais relevantes do projeto armazenados, para servirem de base para futuros projetos;
- Manter a rastreabilidade dos requisitos do projeto, e controle total do seu escopo projeto;

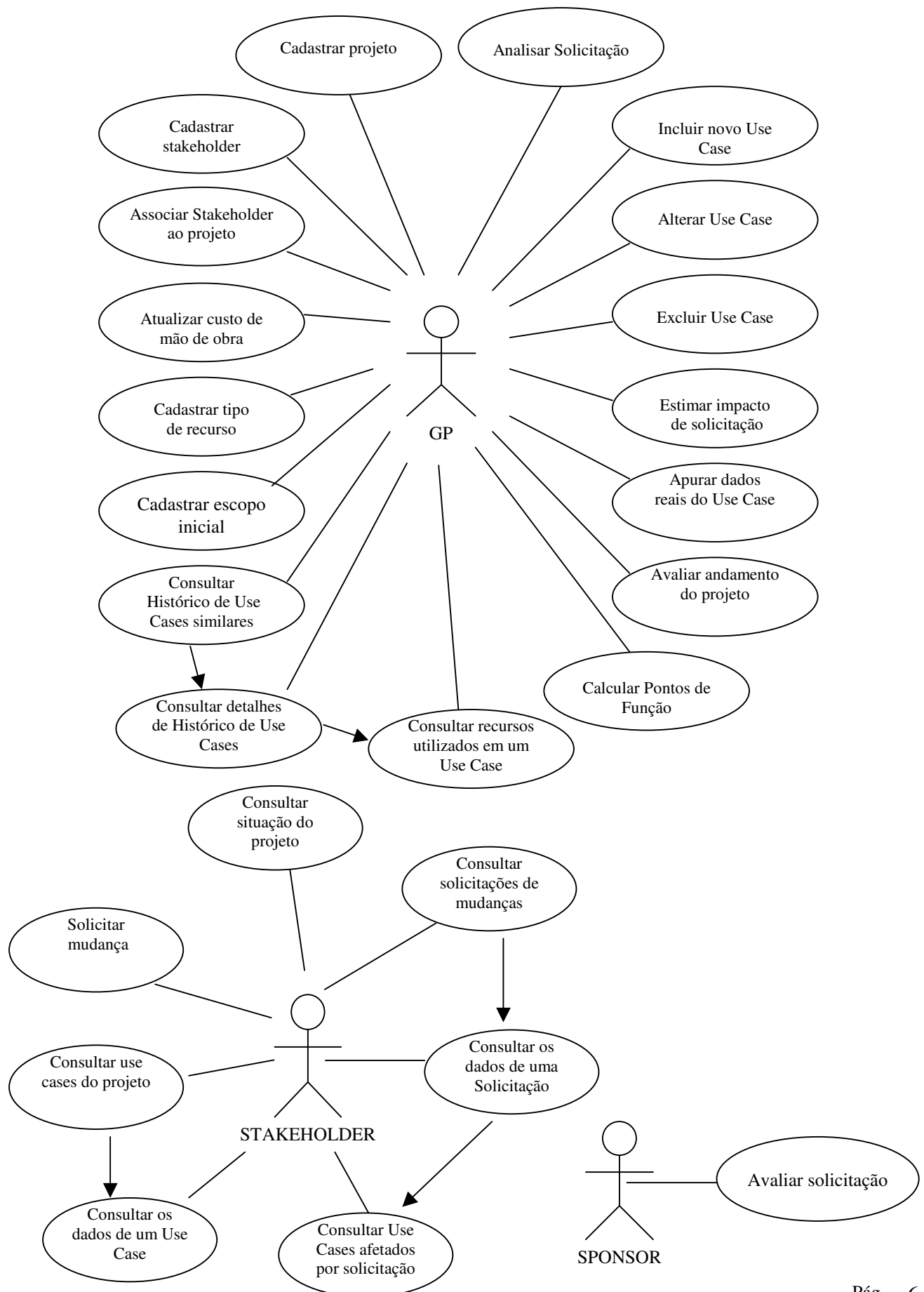
##### **IMPORTANTES:**

- Conhecer os stakeholders do projeto, com seus níveis de competência, quanto à participação no projeto e permissão de solicitação de mudança e/ou aprovação das funções do projeto;
- Manter controle dos custos realizados no projeto e de forma clara, preferencialmente gráfica, apresentar um comparativo com o custo previsto;

##### **DESEJÁVEIS:**

- Controlar as mudanças solicitadas e não aprovadas, com as devidas justificativas;
- Permitir aos stakeholders autorizados, o registro de solicitações de mudanças em funções previamente definidas;
- Permitir consultas sobre a situação do projeto, acessível a todos os stakeholders.

### 3.2 – DIAGRAMA DE CASOS DE USO



### 3.3 – DETALHAMENTO DOS CASOS DE USO

#### 3.3.1 - UC01 – Cadastrar Projeto

**ATORES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** O GP cadastra um projeto a ser controlado pelo sistema, informando sua descrição, data de início e previsões de data de término e custo total.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 – Popula um combo com os códigos dos projetos cadastrados;
2 - O GP insere o Código do projeto a ser incluído e os dados correspondentes e clica no ícone Atualizar;	
	3 - Armazena os dados do projeto .

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O GP deseja alterar ou excluir um projeto já cadastrado.

AÇÃO: O GP seleciona no combo o código do projeto e clica no ícone Buscar;  
O sistema apresenta os dados do projeto;  
Para alterar, o GP altera os dados desejados e clica no ícone Atualizar;  
Para excluir, O GP e clica no ícone Excluir e confirma, após solicitação do sistema.

CONDIÇÃO: O GP tenta incluir um projeto já cadastrado.

AÇÃO: Apresenta mensagem “Já existe projeto cadastrado com esse código”

CONDIÇÃO: O GP tenta excluir um projeto com dados de acompanhamento informados.

AÇÃO: Apresenta mensagem “Não é possível excluir esse projeto. Existem registros associados”

CONDIÇÃO: Ao clicar em Atualizar um ou mais campos da tela não foram informados.

AÇÃO: Apresenta mensagem “PREENCHIMENTO OBRIGATÓRIO” e posiciona o cursor no primeiro campo não preenchido.

CONDIÇÃO: O GP deseja apagar todos os dados da tela.

AÇÃO: O GP clica no ícone Limpar.

CONDIÇÃO: O GP deseja sair da aplicação sem efetivar a gravação.

AÇÃO: O GP clica no ícone Sair.

### 3.3.2 - UC02 – Cadastrar Stakeholder

**ATORES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** O GP cadastra um Stakeholder de um projeto.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Apresenta o código, Nome e setor de todos os Stakeholders cadastrados;
2 - O GP clica no ícone Abrir Linha;	
	3 - Posiciona o cursor na primeira linha vazia;
4 - O GP insere uma ou mais linhas no grid e clica no ícone Atualizar;	
	5 - Para cada linha incluída o sistema gera um código para o stakeholder, na seqüência do maior código cadastrado, apresentando-os a seguir e inclui os dados informados.

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O GP deseja alterar um dado do stakeholder.

AÇÃO: O GP altera os dados desejados e clica no ícone Atualizar;  
O sistema altera os dados do Stakeholder.

CONDIÇÃO: O GP deseja excluir um stakeholder cadastrado.

AÇÃO: O GP dá duplo clique na frente da linha a ser excluído e clica no ícone Atualizar;  
O sistema exclui o Stakeholder.

CONDIÇÃO: O GP inclui um Stakeholder já está cadastrado no sistema.

AÇÃO: O sistema retorna mensagem “STAKEHOLDER JÁ CADASTRADO” e posiciona o cursor no nome correspondente.

CONDIÇÃO: O GP exclui um Stakeholder que está associado a um projeto.

AÇÃO: Apresenta mensagem “Não é possível excluir esse projeto. Existem registros associados”

CONDIÇÃO: O GP deseja sair da aplicação sem efetivar a gravação.

AÇÃO: O GP clica no ícone Sair.

### 3.3.3 - UC03– Associar Stakeholder ao Projeto

**ATORES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** O GP associa um Stakeholder previamente cadastrado a um projeto, indicando a atividade que o mesmo exercerá no referido projeto.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula um combo com os códigos dos projetos não encerrados, outro com o nome e setor de cada stakeholder cadastrado e um terceiro com as atividades cadastradas;
2 - O GP seleciona o código do projeto;	
	3 - Apresenta o nome do projeto;
4 - O GP seleciona o stakeholder e sua atividade no projeto e clica no ícone Atualizar;	
	5 - Registra a associação indicada;

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O GP deseja excluir uma associação cadastrada.

AÇÃO: O GP clica no ícone Buscar;  
O sistema apresenta os demais dados do projeto;  
O GP dá duplo clique na frente da linha que deseja excluir e clica no ícone Excluir;  
O sistema exclui a associação indicada;

CONDIÇÃO: Ao clicar em Atualizar o stakeholder já está associado ao sistema selecionado.

AÇÃO: O sistema retorna mensagem “STAKEHOLDER JÁ ASSOCIADO A ESSE PROJETO” .

CONDIÇÃO: O GP exclui a associação de um Stakeholder a um projeto com dados registrados.

AÇÃO: Apresenta mensagem “Não é possível excluir esse projeto. Existem registros associados”

CONDIÇÃO: O GP deseja mudar para outros projetos cadastrados.

AÇÃO: O GP clica no ícone de Navegação correspondente.

CONDIÇÃO: O GP deseja desfazer seleções efetuadas para reiniciar o projeto sem atualizar.

AÇÃO: O GP clica no ícone Limpar.

CONDIÇÃO: O GP deseja sair da aplicação sem efetivar a gravação.

AÇÃO: O GP clica no ícone Sair.

### 3.3.4 - UC04 – Atualizar Custo da Mão de Obra

**ATORES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** Permite atualização do valor hora de mão de obra dos recursos de desenvolvimento de um projeto.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula o combo com as descrições dos tipos de recursos cadastrados;
2 - O GP seleciona um dos recursos apresentados;	
	3 - Apresenta em um grid os dados de cada atualização de valor do recurso selecionado em ordem cronológica;
4 - O GP clica no ícone Abrir Linha;	
	5 - Posiciona o cursor na primeira linha vazia do grid;
6 - O GP insere uma ou mais linhas do grid apresentado com a data de início de vigência e o novo valor hora do referido recurso e clica no ícone Atualizar;	
	7 - Registra o novo valor, juntamente com o user-id e data de atualização;

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O GP altera um valor apresentado na tabela e clica em Atualizar.

AÇÃO: O sistema retorna mensagem “Não é possível alterar registros desta tabela” .

CONDIÇÃO: Ao Atualizar o recurso indicado já consta no sistema com data igual à informada.

AÇÃO: O sistema retorna mensagem “Valor já existente com data igual” .

CONDIÇÃO: O GP deseja sair da aplicação sem efetivar a gravação.

AÇÃO: O GP clica no ícone Sair.

### 3.3.5 - UC05 – Cadastrar Tipo de Recurso

**ATORES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** Cadastra os Tipos de Recursos possíveis no Desenvolvimento dos projetos.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula um grid com os dados dos tipos de recursos cadastrados;
2 - O GP clica no ícone Abrir Linha;	
	3 - Posiciona o cursor na primeira linha vazia do grid;
4 - O GP insere uma ou mais linhas do grid apresentado e clica no ícone Atualizar;	
	5 - Inclui os dados informados, juntamente com a chave de acesso e data de atualização;

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O GP deseja alterar um Tipo de Recurso cadastrado.

AÇÃO: O GP altera os dados desejados e clica no ícone Atualizar;

CONDIÇÃO: O GP deseja excluir um Tipo de Recurso cadastrado.

AÇÃO: O GP dá duplo clique na frente da linha que deseja excluir e clica no ícone Excluir;

CONDIÇÃO: Ao clicar em Atualizar o tipo de recurso incluído já consta no sistema.

AÇÃO: O sistema retorna mensagem “Tipo de Recurso já cadastrado” .

CONDIÇÃO: O GP deseja sair da aplicação sem efetivar a gravação.

AÇÃO: O GP clica no ícone Sair.

### 3.3.6 - UC06 – Cadastrar Escopo Inicial

**ATORES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** O GP cadastra Use Cases originais do projeto, levantados, consensados e devidamente aprovados ao final da fase do Anteprojeto.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
1 - O GP obtém a aprovação do Sponsor para o Anteprojeto e pretende cadastrar os Use Cases originais do escopo;	
	2 - Popula um combo com os projetos não encerrados;
3 - O GP seleciona um dos projetos;	
	4 - Apresenta o Nome do projeto e gera um código de Use Case na seqüência do código existente no controle do maior utilizado, apresentando-o no campo apropriado;
5 - O GP informa o nome do Use Case, sua Descrição, a estimativa de Pontos de Função podendo utilizar o recurso de uma tela de cálculo e indica os recursos a serem alocados com as respectivas horas estimadas e clica no ícone Atualizar;	
	6 - Apresenta o Custo de Mão de Obra estimado a partir do valor cadastrado vigente para o recurso correspondente multiplicado pela quantidade de horas informada;

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: Ao clicar em Atualizar um ou mais campos da tela não foram informados.

AÇÃO: O sistema retorna mensagem “PREENCHIMENTO OBRIGATÓRIO” e posiciona o cursor no primeiro campo não preenchido.

CONDIÇÃO: O GP clica no ícone Calculadora de Pontos de Função sem ter informado o Use Case.

AÇÃO: O sistema apresenta mensagem “Deve ser informado Projeto e Use Case”.

CONDIÇÃO: O GP deseja sair da aplicação sem efetivar a gravação.

AÇÃO: O GP clica no ícone Sair.



### 3.3.7 - UC07 – Solicitar Mudança

**ATOES:** Stakeholder

**DESCRIÇÃO:** Um Stakeholder cadastra uma Solicitação de Mudança do projeto. Essa solicitação somente será incorporada ao sistema após uma análise e estimativa de impacto do Gerente do Projeto e a devida aprovação do Sponsor.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
1 - Um stakeholder identifica uma necessidade de alteração em um projeto, por perceber uma falha de definição, uma oportunidade de melhoria ou mesmo devido à mudanças na lei ou nos procedimentos ;	
	2 - Popula um combo com o código dos projetos não encerrados;
3 - O stakeholder seleciona um dos projetos;	
	4 - Apresenta o Nome do projeto e gera um código de Solicitação na seqüência do maior código existente para o referido projeto, apresentando-o no campo apropriado;
	5 - Popula o combo com o Nome e setor dos stakeholders cadastrados e associados ao projeto;
6 - O stakeholder informa a Descrição e a justificativa da Solicitação e clica no ícone Atualizar;	
	7 - Grava os dados digitados juntamente com a data atual.

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: Ao clicar em Atualizar o combo de identificação do stakeholder solicitante não foi informado.

AÇÃO: O sistema retorna mensagem “O stakeholder não foi informado”.

CONDIÇÃO: Ao clicar em Atualizar um ou mais campos da tela não foram informados.

AÇÃO: O sistema retorna mensagem “O campo não aceita Nulo” e posiciona o cursor no primeiro campo não preenchido.

CONDIÇÃO: O stakeholder deseja sair da aplicação sem efetivar a gravação.

AÇÃO: O stakeholder clica no ícone Sair.

### 3.3.8 - UC08 – Analisar Solicitação

**ATORES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** Após o cadastramento de uma solicitação de mudança, cabe ao Gerente do Projeto efetuar uma análise dessa solicitação e registrar inclusões, alterações e/ou exclusões de Use Cases para o projeto.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula um combo com o código dos projetos que possuem solicitações pendentes;
2 - O GP seleciona um dos projetos;	
	3 - Apresenta o Nome do projeto e popula o combo com o Números das Solicitações pendentes para o projeto indicado;
4 - O GP seleciona uma das Solicitações;	
	5 - Apresenta o Nome e setor do solicitante, a Data e a Descrição da Solicitação;
6 - O GP digita a Análise efetuada referente à Solicitação e clica em uma das opções apresentadas nos botões abaixo;	
	7 - Caso tenha sido clicado no botão CRIAR NOVO USE CASE chamar UC09.
	8 - Caso tenha sido clicado no botão ALTERAR USE CASE chamar UC10.
	9 - Caso tenha sido clicado no botão EXCLUIR USE CASE chamar UC11.
10 – Ao encerrar o GP clica no ícone Atualizar.	

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: Ao clicar no botão de alteração ou exclusão de Use Cases, existe outra proposta de mudança ainda pendente para o mesmo Use Case .

AÇÃO: O sistema retorna mensagem “USE CASE COM OUTRA SOLICITAÇÃO ABERTA. PROVIDENCIE ANTES SUA AVALIAÇÃO”.

CONDIÇÃO: O GP deseja sair da aplicação sem efetivar a gravação.

AÇÃO: O GP clica no ícone Sair.

CONDIÇÃO: O GP clica no botão CANCELAR SOLICITAÇÃO.

AÇÃO: O sistema apresenta uma tela de confirmação e ao ser confirmado cancela a Solicitação.

### 3.3.9 - UC09 – Incluir novo Use Case

**ATOES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** O GP registra a inclusão de um novo Use Case após análise de solicitação de mudança do projeto. Esse novo Use Case ficará armazenado separadamente, aguardando a avaliação do Sponsor para ser oficializado ou não no projeto.

#### SEQUÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
1 – O GP conclui a análise de uma solicitação e clica no botão CRIAR NOVO USE CASE;	
	2 - Apresenta o Código do projeto, o Número da Solicitação informados no UC08 e gera um código de Use Case na sequência do código existente no controle do maior utilizado, apresentando-o no campo apropriado;
3 - O GP informa o nome do Use Case, sua Descrição, a estimativa de Pontos de Função podendo para tal utilizar o recurso de uma tela de cálculo e clica no ícone Atualizar;	
	4 - Registra os dados informados com tipo de alteração = INCLUSÃO como proposta de mudança e retorna ao UC08;

#### SEQUÊNCIA ALTERNATIVA:

CONDIÇÃO: Ao clicar em Atualizar um ou mais campos da tela não foram informados.

AÇÃO: O sistema retorna mensagem “PREENCHIMENTO OBRIGATÓRIO” e posiciona o cursor no primeiro campo não preenchido.

CONDIÇÃO: O GP deseja utilizar o recurso de Cálculo de Pontos de Função disponível no sistema e clica no ícone Calculadora de PF.

AÇÃO: O sistema aciona o Use Case UC25-Calcular Pontos de Função.

CONDIÇÃO: O GP deseja sair da aplicação sem efetivar a gravação.

AÇÃO: O GP clica no ícone Sair.

### 3.3.10 - UC10 – Alterar Use Case

**ATORES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** O GP registra a alteração a ser efetuada em um Use Case existente após análise de solicitação de mudança do projeto, visando o seu atendimento. Essa alteração no Use Case ficará armazenada separadamente, aguardando a avaliação do Sponsor para ser oficializada ou não no projeto.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
1 - O GP conclui a análise de uma solicitação e clica no botão ALTERAR USE CASE;	
	2 - Apresenta o Código do projeto, o Número da Solicitação informados no UC08 e popula um combo com os códigos dos Use Cases aprovados para o projeto apresentado.
3 - O GP seleciona o código do Use Case, a ser alterado;	
	4 - Apresenta o Nome, a Descrição e a Quantidade de Pontos de Função prevista para o Use Case;
5 - O GP pode alterar o nome, a descrição, a estimativa de Pontos de Função, podendo para tal utilizar o recurso de uma tela de cálculo e clica no ícone Atualizar;	
	6 - Registra os dados informados com tipo de alteração = ALTERAÇÃO como proposta de mudança e retorna ao UC08.

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: Ao clicar em Atualizar um ou mais campos da tela não foi deixado em branco.

AÇÃO: O sistema retorna mensagem “PREENCHIMENTO OBRIGATÓRIO” e posiciona o cursor no primeiro campo não preenchido.

CONDIÇÃO: O GP deseja utilizar o recurso de Cálculo de Pontos de Função disponível no sistema e clica no ícone Calculadora de PF.

AÇÃO: O sistema aciona o Use Case UC25-Calcular Pontos de Função.

CONDIÇÃO: O GP deseja sair da aplicação sem efetivar a gravação.

AÇÃO: O GP clica no ícone Sair.

### 3.3.11 - UC11 – Excluir Use Case

**ATORES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** O GP registra a exclusão de um Use Case existente após análise de solicitação de mudança do projeto, visando o seu atendimento. Essa exclusão de Use Case ficará armazenada separadamente, aguardando a avaliação do Sponsor para ser oficializada ou não no projeto.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
1 - O GP conclui a análise de uma solicitação e clica no botão EXCLUIR USE CASE;	
	2 - Apresenta o Código do projeto, o Número da Solicitação informados no UC08 e popula um combo com os códigos dos Use Cases aprovados para o projeto apresentado.
3 - O GP seleciona o código do Use Case a ser excluído;	
	4 - Apresenta o Nome, a Descrição e a Quantidade de Pontos de Função prevista para o Use Case;
5 - O GP clica no ícone Atualizar;	
	6 - Registra os dados informados com tipo de alteração = EXCLUSÃO como proposta de mudança e retorna ao UC08.

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O GP deseja sair da aplicação sem efetivar a gravação.

AÇÃO: O GP clica no ícone Sair.

### 3.3.12 - UC12 – Estimar impacto de Solicitação

**ATORES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** O GP registra estimativa de impacto da solicitação no projeto quanto à quantidade de horas de esforço estimada para cada tipo de esforço, mudança do prazo final do projeto e custo do desenvolvimento, após análise e estimativa análoga efetuada com o auxílio do UC21.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula o combo Cód.Projeto com os projetos que tenham solicitações analisadas e ainda sem estimativa de impacto;
2 - O GP seleciona o código do Projeto;	
	3 - Popula o combo Num.Use Case com os Use Cases do projeto selecionado, que tenham Solicitações analisadas;
4 - O GP seleciona o código do Use Case;	
	5 - Popula o combo Num.Solicitação correspondente ao Use Case selecionado;
6 - O GP seleciona os recursos previamente populados e preenche a quantidade de horas, previstas para a solicitação;	
	7 - Ao tirar o foco da quantidade de horas, o sistema acessa o valor hora atual do recurso selecionado, multiplica-o pela quantidade de horas e apresenta no valor da Mão de Obra correspondente, totalizando abaixo juntamente com a quantidade de horas;
8 - O GP preenche as Variações de Duração e do Custo e clica em Atualizar;	
	9 - Grava a quantidade de horas e o custo de cada tipo de recurso relacionado, juntamente com a data atual e a variação da duração e do custo, altera a situação da solicitação e retorna ao UC08.

#### SEQÜÊNCIA ALTERNATIVA:

**CONDIÇÃO:** O GP clica em Atualizar sem selecionar algum Recurso, ou sem o preenchimento da quantidade de horas ou da Variação de Duração ou de Custo..

**AÇÃO:** Retorna a mensagem “PREENCHIMENTO INCOMPLETO”;

**CONDIÇÃO:** O GP deseja sair da aplicação sem efetivar a gravação.

**AÇÃO:** O GP clica no ícone Sair.

### 3.3.13 - UC13 – Avaliar Solicitação

**ATOES:** Sponsor

**DESCRIÇÃO:** Apresenta justificativa e estimativas de impacto de uma solicitação de mudança, apoiando a decisão do Sponsor quanto à sua aprovação ou rejeição. Se o Sponsor confirmar a aprovação da solicitação, estará assumindo as novas estimativas apresentadas para o projeto.

#### SEQUÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula o combo Cód.Projeto com os projetos que possuem solicitações com estimativa de impacto e sem avaliação;
2 - O Sponsor seleciona o código do Projeto;	
	3 - Apresenta o Nome do projeto e popula o combo Num.Solicitação com as solicitações do projeto selecionado que têm solicitações aguardando validação;
4 - O Sponsor seleciona o Número da Solicitação;	
	5 - Apresenta o nome e setor do solicitante, a data, descrição, justificativa e análise da solicitação, a quantidade de dias adicionais e soma dos custos adicionais e de mão de obra de cada recurso dos Use Cases da solicitação, a data prevista de implantação correspondente à previsão atual + a quantidade de dias adicionais apresentada e o custo total, correspondente ao Custo atual + a Variação do Custo apresentada.
6 - O Sponsor analisa os dados apresentados e clica no botão Aprovar ou no botão Rejeitar;	
	7 - Altera a data da avaliação com a data atual, a situação para “Aprovada” ou “Rejeitada” de acordo com a definição do Sponsor;
	8 - Se a solicitação foi rejeitada, apresenta caixa de diálogo para preenchimento da Justificativa;
9 - Se a solicitação foi rejeitada, o Sponsor informa a justificativa da rejeição;	
	10 - Se a solicitação foi rejeitada, atualiza a justificativa na solicitação;
	11 - Se a solicitação foi aprovada, oficializa os dados propostos do Use Case e respectivos recursos com tipo de informação = previsto. Inclui, altera ou exclui o Use Case, de acordo com o tipo de alteração do Use Case.

### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: Ao ser acionada a aplicação, não consta solicitação a ser avaliada.

AÇÃO: Retorna “NÃO CONSTAM SOLICITAÇÕES PENDENTES DE AVALIAÇÃO”;

CONDIÇÃO: O Sponsor clica em Cancelar na Caixa de Diálogo da Justificativa de Rejeição.

AÇÃO: O sistema retorna permitindo a mudança para Aprovar sem apagar os dados da tela e sem ter atualizado os arquivos;

CONDIÇÃO: O Sponsor clica em Novo.

AÇÃO: O Sistema limpa todos os campos e popula apenas o combo do projeto conforme o segundo item;

CONDIÇÃO: O Sponsor deseja sair da aplicação sem efetivar a gravação.

AÇÃO: O Sponsor clica no ícone Sair.

### 3.3.14 - UC14 – Apurar dados reais do Use Case

**ATOES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** Após a implantação do sistema ou de um módulo do sistema, o GP informa os dados reais coletados no desenvolvimento dos Use Cases concluídos do projeto.

### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula um combo com os códigos dos projetos que possuam Use Cases sem dados reais apurados;
2 - O GP seleciona o projeto ao qual pertence o Use Case a ser atualizado;	
	3 - Apresenta o Nome do projeto selecionado e popula um combo com os códigos dos Use Cases sem dados reais apurados;
4 - O GP seleciona o código do Use Case;	
	5 - Apresenta o Nome do Use Case selecionado;
6 - O GP informa a Quantidade de Pontos de Função e para cada recurso utilizado, informa o Tipo de Recurso e a Quantidade de Horas de Esforço do Recurso no desenvolvimento;	
	7 - Obtém o valor da mão de obra dos Recursos de Desenvolvimento;
8 - O GP clica em Atualizar;	
	9 - Registra os dados informados e a data atual.

### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O GP deseja sair da aplicação sem efetivar a gravação.

AÇÃO: O GP clica no ícone Sair.



### 3.3.15 - UC15 – Avaliar andamento do projeto

**ATOES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** Registra periodicamente a avaliação do andamento do desenvolvimento do projeto.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula um combo com os códigos dos projetos não encerrados;
2 - O GP seleciona o projeto a ser avaliado;	
	3 - Apresenta o Nome, a data de atualização, as datas de início e de término do projeto, o percentual de tempo transcorrido (calculado) e o Custo total e a situação do projeto;
4 - O GP informa o percentual de conclusão do projeto até o momento;	
	5 - Calcula a data de projeção de término em uma Regra de Três entre os percentuais de tempo transcorrido, percentual de conclusão e data de término previsto. Se o % Conclusão estimado for maior que o % Tempo transcorrido assumir a situação Atrasado, mantendo os ícones correspondentes inibidos;
6 - O GP informa a Variação de Custo, clica em um dos botões correspondentes à Situação Atual do Projeto (caso não esteja em atraso) e os respectivos motivos das variações de Custo e Duração e posteriormente clica no ícone Atualizar;	
	7 - O sistema registra os dados informados e a data atual e apresenta as novas estimativas de Custo e duração total do Projeto. Se o percentual de conclusão for = 100%, altera situação do projeto para “Encerrado”, senão alterar para a Situação informada.

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O GP clica em Atualizar sem ter preenchido um dos campos abertos.

AÇÃO: Retorna mensagem “PREENCHIMENTO OBRIGATÓRIO” e posiciona o cursos no primeiro campo irregular.

CONDIÇÃO: O GP deseja sair da aplicação sem efetivar a gravação.

AÇÃO: O GP clica no ícone Sair.

### 3.3.16 - UC16 – Consultar Use Cases do projeto

**ATORES:** Stakeholder

**DESCRIÇÃO:** Stakeholder consulta os Use Cases pertencentes a um projeto.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula um combo com os códigos de todos os projetos cadastrados;
2 - O Stakeholder seleciona o projeto que deseja consultar;	
	3 - Apresenta o nome do projeto e para cada Use do projeto informa o seu código, o nome e situação. Totaliza a quantidade de Use Cases aprovados e propostos;

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O stakeholder dá duplo clique em uma linha de Use Case apresentado.

AÇÃO: Aciona o Use Case 17.

CONDIÇÃO: O stakeholder deseja sair da aplicação sem efetivar a consulta.

AÇÃO: O stakeholder clica no ícone Sair.

### 3.3.17 - UC17 – Consultar os dados de um Use Case

**ATOES:** Stakeholder

**DESCRIÇÃO:** Stakeholder consulta um determinado Use Case de um projeto. Esta consulta pode ser efetuada diretamente através do Menu, sendo que neste caso informa o Código do projeto e do Use Case. Pode também ser acessada através dos Use Cases 16 ou 20, dando duplo clique na frente da linha de um determinado Use Case, assumindo automaticamente sua identificação.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula um combo com os códigos de todos os projetos cadastrados, caso tenha sido acionado pelo menu. Se acionada por outra tela, apresenta o código do projeto da tela anterior;
2 - O Stakeholder seleciona o projeto que deseja consultar, caso tenha sido acionado pelo menu;	
	3 - Apresenta o Nome do projeto e popula um combo com os códigos dos Use Cases do projeto selecionado, caso tenha sido acionado pelo menu. Se acionada por outra tela, apresenta o código do Use Case da tela anterior;
4 - O Stakeholder seleciona o Use Case que deseja consultar, caso tenha sido acionado pelo menu;	
	5 - Apresenta os seguintes dados do Use Case selecionado: <ul style="list-style-type: none"><li>• descrição do Use Case;</li><li>• data de aprovação do Use Case;</li><li>• tamanho em Pontos de Função previsto;</li><li>• tamanho em Pontos de Função real;</li><li>• Custo da Mão de Obra previsto;</li><li>• Custo da Mão de Obra real.</li></ul>

#### SEQÜÊNCIA ALTERNATIVA:

**CONDIÇÃO:** Após ter entrado nesta aplicação pelo menu, o GP deseja sair da aplicação sem completar a consulta.

**AÇÃO:** O stakeholder clica no ícone Sair.

### 3.3.18 - UC18 – Consultar solicitações de mudanças

**ATORES:** Stakeholder

**DESCRIÇÃO:** Stakeholder consulta Solicitações pertencentes a um projeto.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula um combo com os códigos de todos os projetos cadastrados;
2 - O Stakeholder seleciona o projeto que deseja consultar;	
	3 - Apresenta o nome do projeto e para cada solicitação do projeto informa o Número, descrição e situação da Solicitação. Totaliza a quantidade de Solicitações listadas;

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O stakeholder dá duplo clique em uma linha de Solicitação.

AÇÃO: Aciona o Use Case 19.

CONDIÇÃO: O stakeholder deseja sair da aplicação sem efetivar a consulta.

AÇÃO: O stakeholder clica no ícone Sair.

### 3.3.19 - UC19 – Consultar os dados de uma solicitação

**ATOES:** Stakeholder

**DESCRIÇÃO:** Stakeholder consulta uma determinada Solicitação de Mudança. Esta consulta pode ser efetuada diretamente através do Menu, sendo que neste caso informa o Código do projeto e o número da solicitação. Pode também ser acessada através do Use Case 18, dando duplo clique na frente da linha de uma determinada Solicitação, assumindo automaticamente a identificação da Solicitação.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula um combo com os códigos dos projetos cadastrados, caso tenha sido acionado pelo menu. Se acionada por outra tela, apresenta o código do projeto da tela anterior;
2 - O Stakeholder seleciona o projeto que deseja consultar, caso tenha sido acionado pelo menu;	
	3 - Apresenta o Nome do projeto e popula um combo com os Números de Solicitações do projeto selecionado, caso tenha sido acionado pelo menu. Se acionada por outra tela, apresenta o Número da Solicitação da tela anterior;
4 - O Stakeholder seleciona o Número da Solicitação que deseja consultar, caso tenha sido acionado pelo menu;	
	5 - Apresenta o nome e setor do solicitante, a data, descrição, justificativa e análise da solicitação, data da análise, justificativa da rejeição, data da avaliação, quantidade de dias adicionais, variação de Custo e Situação da Solicitação.

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O stakeholder clica no botão USE CASES AFETADOS

AÇÃO: Aciona o Use Case 20.

CONDIÇÃO: Após ter entrado nesta aplicação pelo menu, o stakeholder deseja sair da aplicação sem completar a consulta.

AÇÃO: O GP clica no ícone Sair.

### 3.3.20 - UC20 – Consultar Use Cases afetados por uma solicitação

**ATORES:** Stakeholder

**DESCRIÇÃO:** Stakeholder consulta os Use Cases associados a uma Solicitação. Esta consulta pode ser efetuada diretamente através do Menu, sendo que neste caso informa o Código do projeto e o número da solicitação. Pode também ser acessada através do Use Case 19, clicando no botão USE CASES AFETADOS, assumindo automaticamente a identificação da Solicitação.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula um combo com os códigos dos projetos que tenha Solicitações de Mudanças, caso tenha sido acionado pelo menu. Se acionada por outra tela, apresenta o código do projeto da tela anterior;
2 - O Stakeholder seleciona o projeto que deseja consultar, caso tenha sido acionado pelo menu;	
	3 - Apresenta o Nome do projeto e popula um combo com os Números de Solicitações do projeto selecionado, caso tenha sido acionado pelo menu. Se acionada por outra tela, apresenta o Número da Solicitação da tela anterior;
4 - O Stakeholder seleciona o Número da Solicitação que deseja consultar, caso tenha sido acionado pelo menu;	
	5 - Apresenta a data da solicitação, a situação e para cada Use Case associado à solicitação informa o código e o nome do Use Case. Totaliza a quantidade de Use Cases listados;

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O stakeholder dá duplo clique em uma linha de Use Case apresentado.

AÇÃO: Aciona o Use Case 17.

CONDIÇÃO: Após ter entrado nesta aplicação pelo menu, o stakeholder deseja sair da aplicação sem efetivar a consulta.

AÇÃO: O stakeholder clica no ícone Sair.

### 3.3.21 - UC21 – Consultar histórico de Use Cases similares

**ATORES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** Apresenta médias dos dados coletados de Use Cases de projetos similares anteriores, com relação aos seus prazos reais.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula um combo com os códigos de todos os projetos não encerrados.
2 - O GP seleciona o projeto que deseja consultar;	
	3 - Popula um combo com os códigos dos Use Cases do projeto selecionado;
4 - O GP seleciona o Use Case que deseja consultar;	
	5 - Através da Quantidade estimada de Pontos de Função do Use Case consultado, busca na base histórica de Use Cases os que se enquadrem na faixa de +- X% de Pontos de Função Real, onde X é um dado parametrizável no sistema.
	6 - Apresenta os tipos de Recursos Utilizados no desenvolvimento dos Use Cases similares com a média de Horas trabalhadas em cada tipo de recurso. Apresenta também o resultado dessa média de horas multiplicada pelo custo atual da hora do recurso. Os totais de horas e Custos também são apresentados.

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O stakeholder clica no botão CONSULTA DETALHES

AÇÃO: Aciona o Use Case 22.

CONDIÇÃO: O stakeholder deseja sair da aplicação sem efetivar a consulta.

AÇÃO: O stakeholder clica no ícone Sair.

### 3.3.22 - UC22 – Consultar detalhes de Histórico de Use Cases

**ATORES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** Apresenta Use Cases similares ao indicado, constantes na base histórica de Use Cases. Pode ser acionada através do Use Case 21, recebendo a identificação do Use Case que servirá de comparação, ou ser acionado diretamente através do Menu.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula um combo com os códigos dos projetos não encerrados, caso tenha sido acionado pelo menu. Se acionada por outra tela, apresenta o código do projeto da tela anterior;
2 - O GP seleciona o projeto que deseja consultar, caso tenha sido acionado pelo menu;	
	3 - Popula um combo com os códigos dos Use Cases do projeto selecionado, caso tenha sido acionado pelo menu. Se acionada por outra tela, apresenta o código do Use Case da tela anterior;
4 - O GP seleciona o Use Case que deseja consultar, caso tenha sido acionado pelo menu;	
	5 - Caso tenha sido acionada pelo Menu, através da Quantidade estimada de Pontos de Função do Use Case consultado, busca na base histórica de Use Cases os que se enquadrem na faixa de +-X% de Pontos de Função Real, onde X é um dado parametrizável no sistema.
	6 - Apresenta a quantidade de Pontos de Função prevista e para cada Use Case similar encontrado, apresenta o Código do Projeto, código do Use Case, Quantidade de P.F. real e Quantidade de horas de esforço real.

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O GP dá duplo clique em uma linha de Use Case apresentado.

AÇÃO: Aciona o Use Case 23.

CONDIÇÃO: Após ter entrado nesta aplicação pelo menu, o GP deseja sair da aplicação sem efetivar a consulta.

AÇÃO: O GP clica no ícone Sair.



### 3.3.23 - UC23 – Consultar recursos utilizados em um Use Case

**ATORES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** Apresenta os recursos utilizados em um Use Case encerrado. Pode ser acionada através do Use Case 22, recebendo a identificação do Use Case que servirá de comparação, ou ser acionado diretamente através do Menu.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula um combo com os códigos dos projetos com Quantidade de Pontos de Função Real informados, caso tenha sido acionado pelo menu. Se acionada por outra tela, apresenta o código do projeto da tela anterior;
2 - O GP seleciona o projeto que deseja consultar, caso tenha sido acionado pelo menu;	
	3 - Popula um combo com os códigos dos Use Cases do projeto selecionado, caso tenha sido acionado pelo menu. Se acionada por outra tela, apresenta o código do Use Case da tela anterior;
4 - O GP seleciona o Use Case que deseja consultar, caso tenha sido acionado pelo menu;	
	5 - Apresenta os Recursos utilizados no desenvolvimento do Use Case e as correspondentes quantidades de Horas de esforço e Custo de Mão de Obra.

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O GP deseja sair da aplicação sem efetivar a consulta.

AÇÃO: O GP clica no ícone Sair.

### 3.3.24 - UC24 – Consultar situação do projeto

**ATOES:** Stakeholder

**DESCRIÇÃO:** Stakeholder consulta Situação de um projeto.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula um combo com os códigos de todos os projetos cadastrados;
2 - O GP seleciona o projeto que deseja consultar;	
	3 - Apresenta o Nome do Projeto, descrição, data de início, situação, percentual de conclusão, custo total e data de conclusão.

#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: O GP deseja sair da aplicação sem efetivar a consulta.

AÇÃO: O GP clica no ícone Sair.

### 3.3.25 - UC25 – Calcular Pontos de Função

**ATORES:** Gerente do Projeto (GP)

**DESCRIÇÃO:** Auxilia o Gerente do projeto na atividade de dimensionar um Use Case em Pontos de Função tanto na fase de Previsão quanto na obtenção do dimensionamento real.

#### SEQÜÊNCIA TÍPICA:

Ação dos Atores	Ação do Sistema
	1 - Popula um combo com os códigos de todos os projetos cadastrados;
2 - O GP seleciona o projeto que deseja consultar;	
	3 - Popula um combo com os códigos de todos os Use Cases do projeto selecionado;
4 - O GP informa os dados pertinentes a cada aba;	
	5 - Calcula a quantidade de Pontos de Função correspondentes às informações dadas;

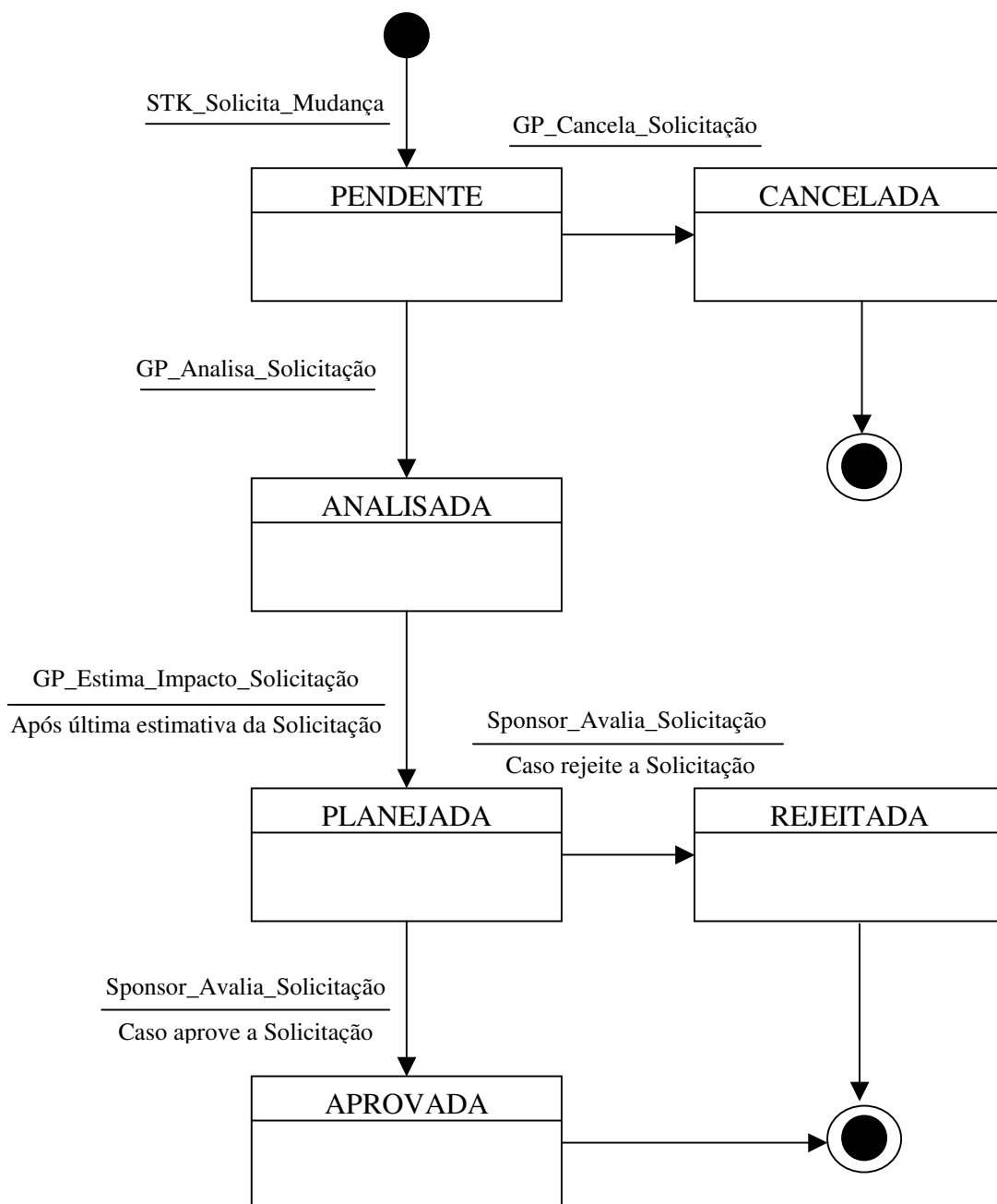
#### SEQÜÊNCIA ALTERNATIVA:

CONDIÇÃO: Caso a função tenha sido acionada por outra tela.

AÇÃO: As identificações de Projeto e Use Case são assumidas automaticamente e após o cálculo, ao sair da função o resultado é transportado para a tela acionadora.

### 3.4 - DIAGRAMA DE TRANSIÇÃO DE ESTADO

#### SITUAÇÃO DA SOLICITAÇÃO DE MUDANÇA



## **4 – REQUISITOS NÃO FUNCIONAIS**

### **4.1 Usabilidade**

Esta seção descreve os requisitos não funcionais associados à facilidade de uso da interface com o usuário, material de treinamento e documentação do sistema.

- **Interface com o usuário**

O sistema deve prover uma interface intuitiva e amigável, especialmente para os stakeholders não habituados com a utilização de sistemas.

**Prioridade:** Importante

- **Apoio ao usuário**

O sistema deve possuir um help com todas as operações possíveis para possibilitar uma rápida consulta às funcionalidades do sistema.

**Prioridade:** Desejável

### **4.2 Segurança**

Esta seção descreve os requisitos não funcionais associados à integridade, privacidade e autenticidade dos dados do sistema.

- **Senhas e acessos**

O sistema deve garantir o acesso somente a pessoas autorizadas. Esse controle é feito por sistema de segurança de acesso independente a este sistema.

**Prioridade:** Importante

### **4.3 Disponibilidade**

Esta seção descreve os requisitos não funcionais associados ao tempo de manutenção das informações para serem acessadas e à disponibilidade das aplicações.

- **Manutenção dos dados**

O sistema deve manter os dados para consultas históricas por tempo indeterminado, não havendo exclusão automática de dados antigos.

**Prioridade:** Importante

- **Aplicações**

O sistema deve estar disponível em horário administrativo de segunda a sexta-feira, e ser utilizado em rede para acesso on-line.

**Prioridade:** Importante

# 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](#)