

**DIEGO BOAVENTURA SILVA
RODRIGO DOURADO SANTOS LOPES**

**ESTIMATIVA DO TAMANHO DE SOFTWARE UTILIZANDO ANÁLISE DE PONTOS
POR FUNÇÃO**

SALVADOR/BAHIA

2008

**DIEGO BOAVENTURA SILVA
RODRIGO DOURADO SANTOS LOPES**

**ESTIMATIVA DO TAMANHO DE SOFTWARE UTILIZANDO ANÁLISE DE PONTOS
POR FUNÇÃO**

Monografia apresentada por Diego Boaventura Silva e Rodrigo Dourado Santos Lopes como requisito parcial para aprovação na disciplina Projeto Final.

Orientador: Luis Morais

SALVADOR/BAHIA

2008

CERTIFICADO

Certifico que a presente memória, ESTIMATIVA DE SOFTWARE A PARTIR DA FASE DE ESPECIFICAÇÃO DE REQUISITOS, UTILIZANDO ANÁLISE DE PONTOS POR FUNÇÃO, foi realizada sob minha direção por DIEGO BOAVENTURA SILVA e RODRIGO DOURADO SANTOS LOPES, constituindo o Projeto Final do Curso de Bacharelado em Informática da Universidade Católica do Salvador – UCSal.

Salvador, 06 de Junho de 2008.

Luis Morais

Curso de Bacharelado em Informática

Universidade Católica do Salvador

RESUMO

Atualmente, os dados e as informações são de grande importância para as organizações, o que evidencia a crescente demanda por novos softwares. Um grande desafio nesse cenário é como estimar o tamanho, custo e esforço do desenvolvimento dessas aplicações de maneira eficiente. Este projeto de pesquisa tem como objetivo apresentar os conceitos ligados a engenharia de software, que tratam das métricas e oferecer a partir daí um protótipo para o auxílio da estimativa do tamanho de um software.

Palavras-chave: Estimativa de Software, Análise de Pontos de Função, Medição, Métrica.

ABSTRACT

Nowadays, data and information are very important for organizations, which highlight the growing demand for new software. A major challenge in this scenario is how estimate the size, cost and effort of developing these applications efficiently. This research project aims to present the concepts related to software engineering, which deal with metrics and from there offer a prototype to aid the estimate size of a software.

Keywords: Estimated Software, Analysis of Points of staff, Measurement, Metric.

SUMÁRIO

1. INTRODUÇÃO	1
2. A ESTIMATIVA DE DESENVOLVIMENTO DE SOFTWARE.....	3
2.1. MEDIDA, MEDIÇÃO E MÉTRICA DE SOFTWARE	3
2.1.1. Métricas orientadas ao tamanho.....	5
2.1.2. Métricas orientadas para função.....	6
3. ANÁLISE DE PONTO DE FUNÇÃO.....	7
3.1. BREVE HISTÓRICO	7
3.2. VISÃO GERAL DA APF	8
3.3. ETAPAS DA APF	10
3.3.1. Determinação do tipo de contagem.....	10
3.3.2. Escopo da contagem e a fronteira da aplicação.....	11
3.3.3. Funções do tipo de dados	12
3.3.3.1. Arquivos lógicos internos (ALI).....	13
3.3.3.2. Arquivos de interface externa (AIE).....	14
3.3.4. Funções do tipo transação.....	17
3.3.4.1. Entradas externas (EE)	18
3.3.4.2. Saídas externas (SE)	18
3.3.4.2. Consultas externas (CE)	19
3.3.5. Fator de ajuste.....	22
3.3.6. Cálculo dos pontos de função ajustados.....	38
3.3.6.1. Projeto de desenvolvimento	38
3.3.6.2. Projeto de melhoria	38
3.3.6.3. Aplicação	39
4. A FERRAMENTA.....	40
4.1. FUNCIONALIDADES	40

4.2. A ARQUITETURA DO APF	42
4.2.1. Camada de modelo.....	43
4.2.1.1. Descrição das Classes e Suas Associações.....	43
4.2.1.2. Persistência de Dados.....	44
4.2.1.3. Base de Dados.....	44
4.2.1.4. Padrão DAO (Data Access Object)	45
4.2.2. Camada de Controle.....	45
4.2.3. Camada de Visualização.....	46
4.3. ESTUDO DE CASO	46
4.3.1. Casos de uso	46
4.3.2. Contagem de Pontos de Função do Estudo de Caso	47
4.3.3. Diagrama de Entidade Relacional (DER).....	52
5. CONCLUSÃO	53
APÊNDICE A – CASOS DE USO.....	54
REFERÊNCIAS BIBLIOGRÁFICAS.....	84

1. INTRODUÇÃO

Nos últimos anos, as organizações têm focado parte de seus esforços em um insumo essencial para sua sobrevivência e desenvolvimento: a informação. É crescente a preocupação com a qualidade e armazenamento de seus dados, com a maneira que eles são coletados, gerenciados e qualificados, tornando-os mais relevantes e importantes. Com isso, surge a necessidade de automação dos processos de gerenciamento desses dados, aumentando à medida que o mercado, os clientes e os processos exigem informações atualizadas e corretas.

Nesse cenário as empresas de desenvolvimento de software, preocupadas em atender essa crescente demanda do mercado, têm se deparado com um considerável crescimento de tamanho e complexidade das aplicações desenvolvidas. Sendo um fator vital para o sucesso desses projetos a correta realização da estimativa do tamanho da aplicação, que implica no custo, na alocação de recurso e tempo do projeto.

As equipes de estimativas, que são de extrema importância, utilizam-se de técnicas e ferramentas além do apoio de modelos de qualidade de software como certificações ISO 9001, CMM, CMMI, MPS. Br para suas estimativas. Todo esse conhecimento e estrutura implicam em utilização de recursos, que às vezes a empresa não dispõe ou não o faz de forma correta, o que vem a ser um problema, pois não existem indicadores de que os projetos serão realizados no tempo contratado e com o custo proposto.

O objetivo desse trabalho é demonstrar os conceitos ligados a engenharia de software, que tratam das métricas, e a partir daí desenvolver um protótipo para o auxílio na estimativa de tamanho da aplicação. Para isso focamos no método de análise de ponto de função, que além de ser amplamente difundido e testado, oferece algumas características desejáveis como a independência de tecnologia, possibilidade de automação e o foco no ponto de vista do usuário.

Para tanto realizamos uma pesquisa exploratória que envolveu a leitura de manuais e artigos relativos à técnica, observando as dificuldades de sua utilização por parte de alguns analistas de sistema e programadores. Além da análise de dados de estimativa de algumas empresas, onde constatamos que em alguns casos

os dimensionamentos do desenvolvimento de seus softwares são realizados em planilhas independentes, sem o acompanhamento e reutilização desses dados.

Esse estudo resultou na construção de um protótipo de uma ferramenta chamada de APF, que realiza essa estimativa e dimensionamento através da técnica de análise de ponto de função, oferecendo como melhoria um gerenciamento unificado, através de uma base histórica única, e um controle sobre a evolução e crescimento do tamanho da aplicação através de um controle de versões.

A continuação deste trabalho divide-se em quatro (4) capítulos, começando pelo segundo capítulo (2) que apresenta a fundamentação teórica, onde são abordados os principais conceitos de medições e métricas de software.

O terceiro capítulo (3) apresenta os conceitos e técnicas da análise de ponto de função.

O quarto capítulo (4) apresenta o desenvolvimento do protótipo, onde foi detalhado todo o processo de especificação e a contagem do tamanho do software.

O quinto capítulo (5) apresenta as considerações finais e uma proposta de trabalho futuro.

2. A ESTIMATIVA DE DESENVOLVIMENTO DE SOFTWARE

Segundo Demarco (1991), em um projeto de software, estimar é uma atividade provocante e de extrema importância, pois o planejamento e o controle de um projeto dependem das estimativas, e estas devem ser o mais correto possível.

Com a estimativa pode-se planejar e controlar mais facilmente o projeto, pois com ela é possível determinar o tamanho do produto a ser construído, o esforço necessário para execução, o custo e prazo de desenvolvimento do projeto.

A partir da manutenção de uma base de dados históricos estimados e realizados dos projetos, a organização pode avaliar a adequação de seu processo de desenvolvimento e extrair indicadores de produtividade e qualidade cada vez mais próximos de sua realidade e, portanto cada vez mais confiáveis. (VASQUEZ, SIMÕES e ALBERT, 2003, p. 144)

2.1. MEDIDA, MEDIÇÃO E MÉTRICA DE SOFTWARE

Os conceitos de medida, medição e métrica de software são definidos a seguir:

Medida é, por definição, a quantificação de uma característica. No caso da área de sistemas devem ser avaliadas não só suas características de produto final, mas também as características dos processos envolvidos em sua concepção e construção. (VASQUEZ, SIMÕES e ALBERT, 2003, p. 32)

Uma medição de software é uma técnica ou método que aplica medidas de software a uma classe de objetos de engenharia de software de forma a atingir um objetivo predefinido. (BASILI, 1989 e BASILI, 1988, p. 431 apud PETERS; PEDRYCZ et al.; 2001.)

“Métrica é uma medição quantitativa simples derivada de qualquer atributo do ciclo de vida do software.” (PETERS e PEDRYCZ et al.; 2001 apud FENTON e KAPOSI, 1987, p. 430)

No desenvolvimento de um projeto devem-se medir as características, propriedades e eventos.

Para identificação do que medir é importante analisar o objetivo do sistema e realizar algumas perguntas que verifique se encaixa com o objetivo, e com isso pode-se determinar uma métrica que quantifique a resposta, tentando ao máximo diminuir a complexidade implicada. Essa complexidade pode ser diminuída agrupando as métricas em módulos com características parecidas, possibilitando a

análise específica de cada uma.

Com os dados encontrados pelas métricas pode-se realizar uma avaliação sobre o ganho em produtividade e qualidade do software, possibilita a criação e o auxílio dos métodos e ferramentas de software para melhora do processo de desenvolvimento, além de uma base fundamental para realização das estimativas.

Pode-se retirar também das métricas, o tamanho ou volume do produto e estabilidade que analisa as mudanças no escopo ou quantidade. Dependendo da variação desses fatores no caso do tamanho para maior e da estabilidade para menor será necessário aumentar os recursos alocados ou aumento do prazo.

As métricas de software, do ponto de vista de medição, podem ser divididas em duas categorias: medidas diretas e indiretas.

As medidas diretas do produto incluem as linhas de código (LOC) produzidas, velocidade de execução, tamanho de memória e defeitos registrados ao longo de certo espaço de tempo. As medidas indiretas do produto incluem funcionalidade, qualidade, complexidade, eficiência, confiabilidade, manutenibilidade. (PRESSMAN, 1995, p. 61).

Pressman (1995) ainda divide em mais categorias o domínio das métricas de software:

- Métrica da produtividade – concentra-se na saída do processo de engenharia de software.
- Métrica da qualidade – oferece uma indicação de quão estreitamente o software conforma-se às exigências implícitas e explícitas do cliente (adequação do software).
- Métrica técnicas – concentram-se nas características do software (por exemplo, complexidade lógica, grau de modularidade) e não no processo por meio do qual o software foi desenvolvido.
- Métricas orientadas ao tamanho – são usadas para compilar as medições diretas da saída e da qualidade da engenharia de software.
- Métricas orientadas para função – oferecem medidas indiretas.
- Métricas orientadas às pessoas – compilam informações sobre a maneira segundo a qual as pessoas desenvolvem software de computador e percepções humanas sobre a efetividade das ferramentas e métodos.

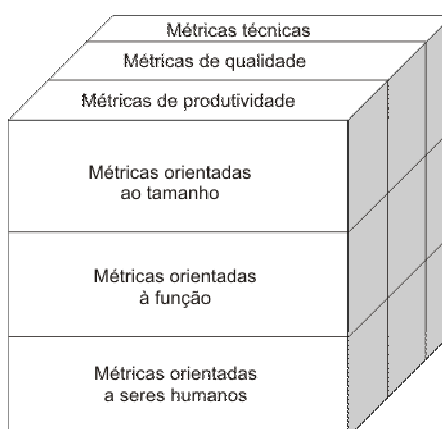


Figura 1 Divisão das métricas em categorias (PRESSMAN, 2000, p. 62)

2.1.1. MÉTRICAS ORIENTADAS AO TAMANHO

“Métricas de software orientadas ao tamanho são medidas diretas de software e do processo por meio do qual ele é desenvolvido.” (PRESSMAN, 1995, p. 62).

A medida direta mais difundida é a de contagem de linhas de código (LOC). Esta técnica é baseada como o próprio nome sugere na contagem das linhas de código do programa em KLOC (mil linhas de código). Com posse desse dado, é possível calcular a produtividade, qualidade, custo e documentação conforme a tabela 1:

Tabela 1 Métricas orientada ao tamanho

Produtividade	Qualidade	Custo	Documentação
KLOC mês	pessoa Defeitos/KLOC	\$/LOC	Paginas de documentação/KLOC

(CORDEIRO, 2000, p.1)

Esta medida é bastante contestada por utilizar as linhas de código como a peça fundamental da métrica. Pois consideram linhas em branco e de documentação sendo que estas não alteram a funcionalidade do software; dependente da linguagem de programação utilizada; da maturidade do programador que pode desenvolver uma função de várias formas e com quantidades diferentes

de linhas; contagem de múltiplos comandos ou declarações em uma única linha como várias linhas, uma para cada comando ou declaração; contagem de uma linha, nos casos em que um único comando ou declaração é expresso em múltiplas linhas, entre outros; desconsideração de delimitadores de blocos de comandos nos casos em que sua utilização é opcional. Sendo assim, seu uso em estimativas fica comprometido, pois o analista deve estipular a quantidade de linhas do projeto, antes mesmo do seu início ou conclusão.

2.1.2. MÉTRICAS ORIENTADAS PARA FUNÇÃO

“A métrica de software orientada à função são medidas indiretas do software e do processo por meio do qual ele é desenvolvido.” (PRESSMAN, 1995, p. 64). As métricas orientadas para função baseiam-se na contagem das funcionalidades, ou seja, utilidades do sistema. O modelo mais utilizado é o de pontos de função (PF).

“Um ponto de função é uma unidade de medida para expressar a quantidade de funcionalidade que um sistema de informação provê a um usuário.” (Wikipédia, 2008, p. 1)

Uma técnica de que utiliza PF é a Complexidade Média, seguindo a versão 5 do *International Software Benchmarking Standards Group* (ISBSG¹). Esta técnica apresenta uma média de complexidade para os projetos de desenvolvimento, distribuídas por tipos de funções, conforme a tabela 2.

Para estimar o tamanho funcional do projeto calcula-se a fórmula abaixo, onde n é o número do tamanho funcional:

$$n = \# EE \times 4,3 + \# SE \times 5,4 + \# CE \times 3,8 + \# ALI \times 7,4 + \# ALE \times 5,5$$

O PF possui uma vantagem em relação ao LOC, pois é independente da linguagem de programação a ser utilizada no desenvolvimento, utiliza dados que são conhecidos logo no início do projeto, tornando assim atrativa para a estimativa.

¹ Organização sem fins lucrativos, cuja missão é ajudar na gestão dos recursos de TI pela melhoria das estimativas de projeto e produtividade, análise de riscos e benchmarking.

Tabela 2 Média de PF não-ajustados, por tipo de função, do ISBSG comparado na tabela de complexidade do IFPUG.

Tipo de Função	Média de PF	IFPUG² (baixo)	IFPUG (médio)	IFPUG (alto)
Arquivo lógico interno (ALI)	7,4	7	10	15
Arquivo lógico externo (ALE)	5,5	5	7	10
Entrada Externa (EE)	4,3	3	4	6
Saída Externa (EE)	5,4	4	5	7
Consulta Externa (CE)	3,8	3	4	6

(VASQUEZ; SIMÕES e ALBERT, Pág. 147)

3. ANÁLISE DE PONTO DE FUNÇÃO

3.1. BREVE HISTÓRICO

Os conceitos sobre a técnica de pontos de função iniciou-se na IBM, através de Allan Albrecht, na década de 70 e veio como uma alternativa para realização das métricas que até então eram feitas através da técnica de linhas de código.

Allan Albrecht tinha que medir a produtividade de vários softwares que foram desenvolvidos pela IBM, mas tinha um grande problema, os softwares foram desenvolvidos em linguagens distintas, o que tornaria o trabalho dele inviável com a utilização de LOC, a partir disso nasceu à idéia da criação de uma medida que fosse independente da linguagem utilizada.

² Entidade sem fins lucrativos, onde promove um melhor gerenciamento dos processos de desenvolvimento de software e manutenção de software, usando PF.

A técnica foi apresentada à comunidade, e logo ganhou vários adeptos na sua utilização, sendo que um grupo de usuários resolveu criar padrões adicionais nas regras de contagem de pontos de função, formando assim o *International Function Point Users Group* ou Grupo Internacional de Usuários de Pontos de Função - IFPUG, sem finalidades lucrativas, com o objetivo de aperfeiçoar o gerenciamento dos processos de construção e manutenção de software, utilizando pontos de função e outras técnicas de medição.

A partir disto apareceram variações da proposta inicial de Allan Albrecht, e em 1990 foi lançada a primeira versão do *Counting Practices Manual* ou Manual de Práticas de Contagem – COM como forma de padronização da técnica e esse padrão é utilizado atualmente.

Hoje em dia, outros métodos estão sendo utilizados para a medição funcional de software, como Mark II e COSMIC-FFP, porém sem grande aceitação no mercado.

No Brasil a utilização da técnica aumentou na década de 90, com apoio da empresa Unisys. A intensificação do uso só veio com a exigência do mercado onde grandes contratos estavam sendo feitos com base em pontos de função.

O *Brazilian Function Point Users Group* – BFPUG é uma célula do IFPUG no Brasil. Criada em 1998, pela evolução de um grupo de usuários do Rio de Janeiro, o FPUG-Rio. Serve um canal de troca de experiências entre profissionais, discussões, agenda de eventos, etc.

3.2. VISÃO GERAL DA APF

Análise de Pontos de Função (APF) é uma técnica para a medição de projetos de desenvolvimento de software, visando estabelecer uma medida de tamanho, em Pontos de Função (PF), considerando a funcionalidade implementada, sob o ponto de vista do usuário. (Wikipédia, 2008, p. 1)

Partindo deste conceito APF, é um processo padronizado para medição de sistemas, onde define o tamanho do software em pontos de função. O estudo das medidas deve ser feita contabilizando as funcionalidades (números de pontos de função) que o software possui, levando a perspectiva do usuário como base.

Tem como objetivos principais medir a funcionalidade que o usuário solicita e recebe medir o desenvolvimento e manutenção de software de forma independente da tecnologia utilizada para a implementação, além disso, deve ser simples o suficiente para minimizar o trabalho adicional envolvido no processo de medição, uma medida consistente entre vários projetos e organizações.

Os benefícios promovidos pela APF são os seguintes:

- Determinar o tamanho de um software ou módulo, através da contagem das funções definidas pelo usuário;
- Ferramenta para ajudar os usuários a determinar os benefícios de um software ou módulo para a sua organização, através da contagem das funções correspondente aos requisitos;
- Auxilia o gerenciamento do projeto no controle do aumento sofrido pelo software no seu desenvolvimento, pois no desenvolvimento o projeto sofre alterações podendo aumentar o escopo do mesmo;
- Ferramenta para medir unidades de software para suportar a análise de produtividade e qualidade;
- Um meio para estimar custo e recursos para desenvolvimento e manutenção de software, com a realização da estimativa pode-se definir custo e recursos necessários para o desenvolvimento do projeto;
- Fator de normalização para comparação de software, de posse das medições é possível comparar softwares ou módulos construídos;
- Embasamento para negociação de contratos justificando os valores do projeto.

Com a contagem de pontos de função é possível determinar algumas fatos para a conclusão do processo de contagem: o tipo de contagem que pode ser de projeto de desenvolvimento, de melhoria ou de aplicação, se o projeto irá conter uma ou mais aplicações ou parte dela, influencia o posicionamento da fronteira da aplicação, o nível do detalhamento da contagem o que facilita futuras contagens.

A contagem dos pontos deve obrigatoriamente obedecer aos seguintes passos propostos e na ordem a seguir:

- 1) Determinar o Tipo da Contagem.
- 2) Identificar o escopo da contagem e a fronteira da aplicação.

- 3) Contar as Funções do Tipo Dados.
- 4) Contar as Funções do Tipo Transação.
- 5) Determinar a contagem dos Pontos de Função Brutos
- 6) Determinar o valor do Fator de Ajuste.
- 7) Calcular o número dos Pontos de Função Ajustados.

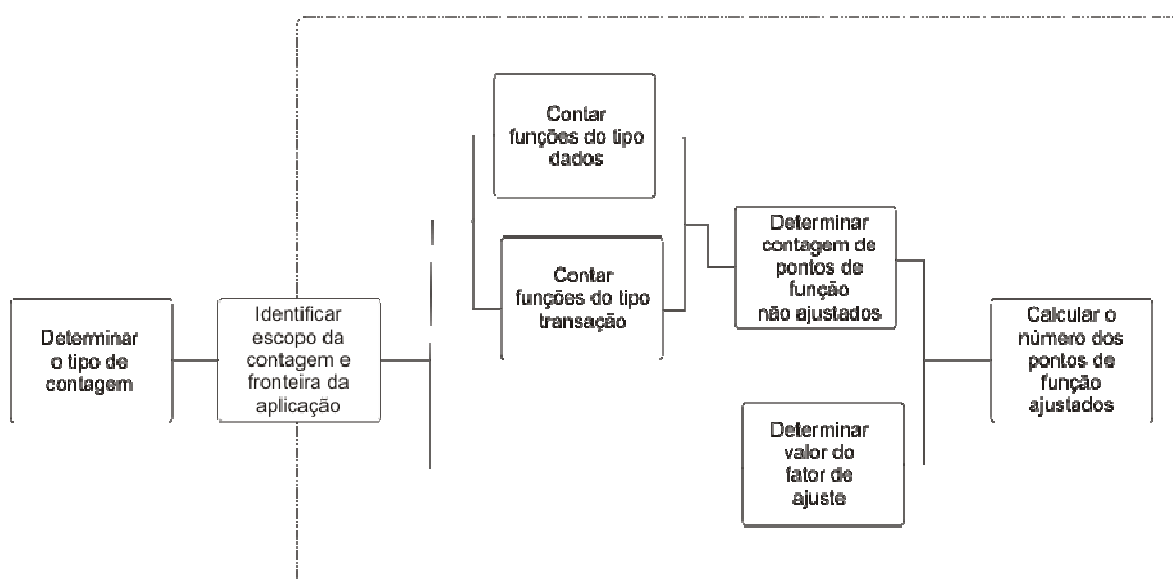


Figura 2 Visão geral do processo de contagem de pontos de função. (VASQUEZ; SIMÕES e ALBERT, p. 20)

3.3. ETAPAS DA APF

3.3.1. DETERMINAÇÃO DO TIPO DE CONTAGEM

A APF inicia-se com a determinação do tipo de contagem. A contagem de pontos de função pode estar vinculada a projetos como a aplicações. São três tipos de contagem, de projeto em desenvolvimento, projeto de melhoria e aplicação ou *baseline*.

Na contagem de projeto de desenvolvimento os números de pontos de função são medidos todas as funcionalidades requeridas pelo usuário desde sua primeira

instalação, contando ainda com conversões de dados que foram realizadas para implantação do projeto.

Na contagem de projeto de melhoria o número de pontos de função mede as funcionalidades adicionadas, modificadas, ou excluídas do sistema, e algumas conversões de dados necessárias.

Na contagem de aplicação o número de pontos de função é medido das funcionalidades informadas pelos usuários existentes na aplicação atual instalada. Esse tipo de contagem é iniciada ao final da contagem de projeto de desenvolvimento e atualizada no final do projeto de contagem de melhoria, pois neste momento todas as funcionalidades já foram definidas pelo usuário, atualizando as funcionalidades do sistema.

Na etapa de Cálculo dos Pontos de Função Ajustados, serão tratadas as fórmulas para cada tipo de contagem.

3.3.2. ESCOPO DA CONTAGEM E A FRONTEIRA DA APLICAÇÃO

A segunda etapa da APF é a determinação do escopo da contagem e da fronteira da aplicação.

Escopo da contagem é onde verifica se o projeto conterà um ou mais sistemas ou parte dele. Podendo estar incluso todas as funcionalidades possíveis, ou parte destas funcionalidades escolhidas pelo usuário, ou ainda específicas como cadastros, gráficos, etc.

“A fronteira da aplicação é a interface conceitual que delimita o software que será medido e o mundo exterior.” (VASQUEZ, SIMÕES e ALBERT, 2003, p. 58)

A Fronteira de uma Aplicação delimita o escopo do software que está sendo medido. O objetivo da determinação da fronteira da aplicação é identificar quais sistemas, aplicações ou subconjuntos de uma aplicação serão medidos.

A Fronteira de uma Aplicação define o que é externo e interno à aplicação e poderá incluir mais de um sistema.

Existem algumas regras para determinação da fronteira da aplicação, tais regras foram especificadas pelo IFPUG:

- A Fronteira é determinada baseando-se na visão do usuário. O foco é no que

pode ser entendido e descrito por ele.

- A Fronteira entre aplicações relacionadas é baseada em separadas áreas funcionais ou módulos sob o ponto de vista de usuário, não em considerações técnicas.
- A contagem de um Projeto de Manutenção inclui todas as funções que estão sendo incluídas, alteradas e excluídas. A Fronteira da aplicação permanece a mesma.

Para facilitar essa definição da fronteira da aplicação, podem-se utilizar as especificações do sistema ou um diagrama de contexto para determinar a fronteira da aplicação; verificar como grupos de dados são mantidos; identificar áreas funcionais pela atribuição de propriedade de certos objetos de análise, como entidades e processos; comparar os critérios utilizados em outras métricas como esforço, duração, custo e defeitos; verificar como a aplicação é gerenciada se é desenvolvida ou mantida na sua totalidade por uma equipe distinta; verificar se o software possui ordens de serviço específicas e independentes.

A determinação da fronteira da aplicação é fundamental para a medição funcional, pois realizada de maneira equivocada, posicionando a fronteira no local errado, pode alterar a perspectiva da contagem de uma visão lógica (princípio da APF) para uma visão física. Podendo ocasionar contagem duplicada da mesma transação por várias aplicações, em vez que cada uma delas contribui com um subprocessamento para o processo elementar do negócio; contagem de funções de transferência de dados entre plataformas ou aplicações; dificuldade na determinação do tipo de transação; duplicidade na contagem de arquivos.

3.3.3. FUNÇÕES DO TIPO DE DADOS

As funções Tipo Dados são representadas pelo agrupamento lógico de dados atualizados ou consultados pela aplicação pontuada, o que permite o atender o usuário a seus requisitos de dados. Esses agrupamentos são chamados Arquivos Lógicos Referenciados e dividem-se em dois tipos: Arquivo Lógico Interno (ALI) e Arquivo de Interface Externo (AIE).

O termo arquivo não representa a definição comum de processamento de dados, como um arquivo de um sistema operacional. Neste caso, a palavra arquivo

refere-se a um agrupamento de dados logicamente relacionados, reconhecido pelo usuário e não a uma implementação física destes grupos.

3.3.3.1. ARQUIVOS LÓGICOS INTERNOS (ALI)

Segundo Vasquez, Simões e Albert (2004) um arquivos lógicos internos (ALI): é um grupo de dados ou informações de controle que o usuário possa identificá-lo possuindo uma relação lógica e estão contidos na fronteira da aplicação.

A principal função é armazenar dados mantidos por meio de um ou mais processos elementares da aplicação sendo contada.

Para identificação de um ALI, é importante analisar se o grupo de dados é lógico e reconhecido pelo usuário e se o grupo de dados é mantido através de pelo menos um processo elementar de dentro da fronteira da aplicação.

Devem ser considerados ALIs	Não são considerados ALIs
Arquivos de Senha mantidos dentro da aplicação;	Repetições do ALI já identificado;
Arquivos de Senha mantidos dentro da aplicação;	Arquivos temporários;
Arquivos de Help mantidos dentro da aplicação;	Arquivos de Ordenação (índices);
Dados relativos a parâmetros mantidos dentro da aplicação;	Arquivos existentes por necessidade tecnológica;
Arquivos de Erros e suas descrições mantidas dentro da aplicação;	Arquivos de Auditoria e/ou Histórico. As informações existentes nesses arquivos são contadas com os arquivos aos quais eles se referem;
Tabelas que armazenam dados mantidos pela aplicação;	Arquivos de backup;

Arquivos de configuração mantidos pela aplicação.	Arquivos gerados para conter informações sobre transações incompletas.
---	--

Quadro 1 Exemplos de ALI (VASQUEZ; SIMÕES e ALBER, 2003, p. 77)

Levando em consideração que arquivos de Senha, help, parâmetros e de erros devem ser contados como ALI se a própria aplicação disponibilizar ao usuário uma forma de atualizá-los. Helps feitos em uma ferramenta externa não devem ser contados como ALI.

Sendo que os consolidados não devem ser contados como ALI se a mesma aplicação mantém os dados bases (dados que servem como base para a geração do consolidado) para que não haja repetição na pontuação. No entanto, se a aplicação não mantém os dados base, o consolidado é pontuado como ALI.

3.3.3.2. ARQUIVOS DE INTERFACE EXTERNA (AIE)

Segundo VASQUEZ, SIMÕES e ALBERT (2004) um arquivo de interface externa (AIE) é um grupo de dados ou informações de controle que o usuário possa identificá-lo onde estão logicamente ligados sendo referenciado pela aplicação.

A função principal é armazenar dados referenciados por meio de um ou mais processos elementares dentro da fronteira da aplicação sendo contada, onde obrigatoriamente um AIE deve ser um ALI de outra aplicação.

A diferença entre ALI e AIE, é que o AIE não é mantido pela aplicação sendo contada.

Para o reconhecimento de um AIE, é importante que o grupo de dados seja lógico e reconhecido pelo usuário, referenciado e externo a aplicação que está sendo contada, onde os dados não devem ser mantidos pela aplicação que está sendo contada, e sim em um ALI de outra aplicação.

Devem ser considerados AIE:	Não devem ser considerados AIE:
Arquivos de Senha mantidos fora da aplicação;	Dados recebidos de outras aplicações que são mantidos em algum ALI da aplicação pontuada;
Arquivos de Help mantidos fora da aplicação	Dados formatados e enviados para fora da fronteira da aplicação;
Dados relativos a parâmetros mantidos fora da aplicação;	Arquivos temporários;
Arquivos de Erros e suas descrições mantidos fora da aplicação.	Arquivos de Ordenação (índices);
	Arquivos existentes por necessidade tecnológica;
	Cópia dos Arquivos Lógicos Externos;
	Arquivos de Auditoria e/ou Histórico. As informações existentes nesses arquivos são contadas com os arquivos aos quais eles se referem.

Quadro 2 Exemplos de AIE (VASQUEZ; SIMÕES e ALBERT,2003, p.78)

Arquivos de Senha, Help, Parâmetros e de Erros devem ser contados como AIE se uma outra aplicação mantiver esses dados, disponibilizando para o usuário uma forma de atualizá-los e se a aplicação pontuada acessar essas informações. Helps feitos em uma ferramenta externa devem ser contados como um AIE de complexidade funcional baixa.

Os consolidados não devem ser contados como AIE se a mesma aplicação consulta os dados bases (dados que servem como base para a geração do consolidado) para que não haja repetição na pontuação. No entanto, se a aplicação não consulta os dados base, o consolidado é pontuado como AIE.

Determinação da Complexidade

Os arquivos lógicos interno e os arquivos de interface externa classificam-se de acordo sua complexidade funcional (baixa, média ou alta) em dois tipos: Número de Tipos de Dados (TD) e Número de Tipos de Registros (TR).

“Um tipo de dado é um campo único, reconhecido pelo usuário, não repetido.” (VASQUEZ; SIMÕES e ALBERT, 2003, p. 81)

Existem algumas regras para a contagem de tipos de dados. Tais regras serão listadas a seguir:

- Contar um tipo de dado para cada campo único reconhecido pelo usuário e não repetido, mantido ou recuperado de um ALI ou AIE através da execução de um processo elementar.
- Quando duas aplicações mantêm ou referenciam o mesmo ALI ou AIE, conte apenas os campos utilizados pela aplicação.
- Contar um tipo de dado para cada campo solicitado pelo usuário para estabelecer um relacionamento com outro arquivo.

“Um tipo de dado de registro é um subgrupo de tipos de dados, reconhecido pelo usuário, componente de um arquivo lógico interno ou arquivo de interface externa.” (VASQUEZ, SIMÕES e ALBERT, 2003, p. 83)

Existem dois tipos de subgrupo:

- Opcionais: são aqueles em que o usuário tem a opção de não informar no processo elementar que cria ou adiciona dados ao arquivo.
- Obrigatórios: são aqueles que o usuário requer que seja sempre utilizado pelo processo elementar que cria ou adiciona dados ao arquivo.

Para contagem dos tipos de registros também existe algumas regras básicas:

- Contar um tipo de registro para cada subgrupo, obrigatório ou opcional, de um ALI ou AIE.
- Se não houver nenhum subgrupo, conte o próprio ALI ou AIE como um tipo de registro.

Para a realização da contagem dos TDs e TRs, classifica-se a complexidade que

está representada pela tabela a seguir:

Tabela 3 Complexidade funcional dos ALI e AIE

Tipos de Registros	Tipos de Dados		
		< 20	20 - 50
1	Baixa	Baixa	Média
2 - 5	Baixa	Média	Alta
> 5	Média	Alta	Alta

(VASQUEZ; SIMÕES e ALBERT, 2003, p. 81)

Depois da contagem da complexidade dos arquivos, é necessário calcular a contribuição utilizando a tabela 4.

Tabela 4 Contribuição dos pontos de função não-ajustados das funções do tipo dado

Tipo de Função	Baixa	Média	Alta
ALI	7 PF	10 PF	15 PF
AIE	5 PF	7 PF	10 PF

(VASQUEZ; SIMÕES e ALBERT, 2003, p. 85)

3.3.4. FUNÇÕES DO TIPO TRANSAÇÃO

“As funções do tipo transação representam a funcionalidade fornecida ao usuário para atender as suas necessidades de processamento de dados pela aplicação.” (VASQUEZ, SIMÕES e ALBERT, 2003, p. 89)

As funções do tipo transação são classificadas em três tipos:

- Entradas Externas (EE);
- Saídas Externas (SE);
- Consultas Externas (CE).

3.3.4.1. ENTRADAS EXTERNAS (EE)

“Uma entrada externa (EE) é um processo elementar que processam dados ou informações de controle que vêm de fora da fronteira da aplicação. A intenção primária de uma EE é manter um ou mais ALLs e/ou alterar o comportamento do sistema.” (Manual IFPUG, 2005, p.71)

São considerados EE	Não são considerados EE
Transações que recebem dados externos na manutenção de Arquivos Lógicos Internos.	Telas de filtro de relatórios e consultas.
Janela que permite adicionar, excluir e alterar registros em arquivos, onde cada ação é uma EE.	Menus.
Processamento em lotes de atualização de bases cadastrais a partir de arquivos em movimento.	Telas de login.

Quadro 3 Exemplos para EE (VASQUEZ; SIMÕES e ALBERT, 2003, p.90)

3.3.4.2. SAÍDAS EXTERNAS (SE)

“Uma saída externa (SE) é um processo elementar que envia dados ou informações de controle para fora da fronteira da aplicação. A intenção primária de uma SE é apresentar informações ao usuário através de lógica de processamento que pode incluir, ou não, a recuperação de dados ou informações de controle. O processamento lógico deve conter pelo menos uma fórmula matemática ou cálculo, criar dados derivados, manter um ou mais ALLs ou alterar o comportamento do sistema.” (Manual IFPUG, 2005, p. 71)

São considerados SE	Não são considerados SE
Relatórios com totalização de dados.	Telas de help.
Relatórios que também atualizam arquivos.	Drop-downs.

Consultas com apresentação de dados derivados ou cálculos.	Consultas e relatórios sem nenhum totalizador, que não atualizam arquivos, não tem dados derivados ou modificam o comportamento do sistema.
Arquivo de movimento gerado para outra aplicação.	
Informações em formato gráfico.	

Quadro 4 Exemplos de SE (VASQUEZ; SIMÕES e ALBERT, 2003, p.91)

3.3.4.2. CONSULTAS EXTERNAS (CE)

“Uma consulta externa (CE) é um processo elementar que envia dados ou informações de controle para fora da fronteira da aplicação. A intenção primária de uma CE é apresentar informações ao usuário através da recuperação de dados ou informações de controle de um ALI ou AIE. O processamento lógico não deve conter fórmulas matemáticas ou cálculos, nem criar dados derivados. Nenhum ALI é mantido durante o processamento e nem o comportamento do sistema é alterado.” (Manual IFPUG, 2005, p. 72)

São considerados CE	Não são considerados CE
Telas de help.	Menus estáticos.
Informações em formato gráfico.	Relatórios que contenham cálculo ou gerem dados derivados.
Drop-downs, desde que recuperem dados de um arquivo, os que possuem os valores codificados no código fonte não são contados.	Consultas que contenham cálculo ou gerem dados derivados.
Telas de logon.	

Menu gerados dinamicamente com base em configuração da aplicação.	
---	--

Quadro 5 Exemplos de CE (VASQUEZ; SIMÕES e ALBERT, 2003, p.91 e 92)

Determinação da Complexidade

Para determinação da complexidade, os EE, SE e CE, são classificados de acordo com sua complexidade funcional podendo ser baixa, média e alta, tendo como base número de arquivos referenciados (AR) e no número de tipos de dados (TD).

Depois disto as quantidades de arquivos referenciados e de tipos de dados, são classificadas de acordo sua complexidade seguindo a tabela abaixo.

Tabela 3 Complexidade funcional para EE

	1 a 4 DER	5 a 15 DER	16 ou mais DER
0 a 1 ALR	Baixa	Baixa	Média
2 ALRs	Baixa	Média	Alta
3 ou mais ALRs	Média	Alta	Alta

(MANUAL IFPUG, 2005, p. 89)

Tabela 4 Complexidade funcional para SE e CE

	1 a 5 DER	6 a 19 DER	20 ou mais DER
0 a 1 ALR	Baixa	Baixa	Média
2 a 3 ALRs	Baixa	Média	Alta
4 ou mais ALRs	Média	Alta	Alta

(MANUAL IFPUG, 2005, p. 90)

Um Arquivo Referenciado (AR) é um arquivo lógico interno lido ou mantido pela função do tipo transação; ou um arquivo de interface externa lido pela função do tipo transação.

Um tipo de dado (TD) é um campo único, reconhecido pelo usuário, não repetido.

Segundo Vasquez, Simões e Albert (2003) existem algumas regras para contagem do AR e do TD, as principais do AR são:

- Contar um arquivo referenciado para cada ALI mantido;
- Contar apenas um arquivo referenciado para cada ALI que seja mantido quanto lido.

Tem outra que não é aplicada para consultas externas:

- Contar um arquivo referenciado para cada ALI ou AIE lido durante o processamento.

Já para os TD as regras principais são:

- Contar um tipo de dado para cada campo, não repetido e reconhecido pelo usuário, que entra ou sai pela fronteira da aplicação e necessário à conclusão do processo;
- Se um campo tanto entra quanto sai pela fronteira da aplicação, deve ser contada uma única vez;
- Os campos que durante o processo elementar são recuperados ou derivados pelo sistema e armazenados em um ALI, mas não atravessam a fronteira da aplicação, não devem ser contados como tipos de dados;
- Contar um único tipo de dado para a capacidade de envio para fora da fronteira da aplicação de uma mensagem de resposta do sistema, indicando um erro verificando durante o processamento, a confirmação da sua conclusão ou a verificação de seu prosseguimento.
- Contar um tipo de dado para a capacidade de especificar uma ação a ser tomada. Mesmo que haja múltiplos meios de ativar o mesmo processo, deve ser contado apenas um tipo de dado.
- Não deve ser contado literal como tipo de dados.
- Não deve ser contado variável de paginação ou campos automáticos gerados pelo sistema.

Concluída a determinação da complexidade das funções do tipo transação, calcular-se sua contribuição para contagem, baseada na tabela 7:

Tabela 5 Contribuição dos pontos de função não-ajustados das funções do tipo transação

Tipo de Função	Baixa	Média	Alta
EE	3 PF	4 PF	6 PF
SE	4 PF	5 PF	7 PF
CE	3 PF	4 PF	6 PF

(VASQUEZ; SIMÕES e ALBERT, 2003, p. 99).

3.3.5. FATOR DE AJUSTE

O Fator de Ajuste é obtido através de uma fórmula que leva em consideração as características gerais do sistema, como por exemplo, interface visual amigável para os usuários, internacionalização, ajuda on-line e outras facilidades, performance, facilidades de manutenção entre outras.

A técnica estabelece um total de 14 Características Gerais do Sistema, listadas a seguir:

1. Comunicação de Dados
2. Processamento Distribuído
3. Performance
4. Configuração Altamente Utilizada
5. Taxa de Transações
6. Entrada de Dados *On-line*
7. Eficiência do Usuário final
8. Atualização *On-line*
9. Processamento Complexo
10. Reutilização

11. Facilidade de Instalação
12. Facilidade de Operação
13. Múltiplos Locais
14. Modificações Facilitadas

Cada uma dessas características é classificada de acordo a tabela de influência, onde seus valores variam de zero a cinco.

Tabela 6 Tabela de Influência

Pontue como	Influência no Sistema
0	Não presente ou sem influência
1	Influência Mínima
2	Influência Moderada
3	Influência Média
4	Influência Significativa
5	Forte influência

(MANUAL IFPUG, 2005, p. 99)

Após determinado os níveis de influência de cada característica, é realizado o ajuste utilizando a seguinte fórmula:

$PF = Contagem_Total * [0,65 + (0,01 * \sum(xn))]$, onde xn varia de 1 a 14, são as respostas das perguntas descritas na figura 3, esse somatório serve como ajuste da complexidade, facilitando e aumentando a precisão na determinação dos pontos de função.

O fator de ajuste, quando aplicado aos pontos de função não-ajustados, pode produzir uma variação de +/- 35%, resultando na contagem final dos pontos de função. Isto implica em um intervalo de variação para o fator de ajuste da ordem de 0,65 a 1,35. O fator de ajuste é responsável pela correção das distorções da etapa anterior.

O fator de ajuste tornou-se opcional no final do ano de 2002, a IFPUG realizou uma pesquisa onde a maioria dos profissionais disse que pararam de utilizar o fator de ajuste.

Foi criado um grupo para o estudo do fator de ajuste, pois a existe uma grande variação de interpretação das características por parte dos membros da IFPUG, como performace e volume de transações, entrada de dados on-line e atualização on-line; e as características estão desatualizadas, já que as características foram inseridas no ano de 1984. E o grupo espera que cheguem a um consenso, deixem como está e continue as 14 características, atualizem ou elimine da técnica de APF o fator de ajuste.

Outros fatores podem ser levados em consideração como: existem requisitos importantes que não fazem parte das 14 características; a variação de +/-35% é insuficiente para representar as funcionalidades gerais da aplicação; o uso do fator de ajuste não traz nenhum benefício adicional aos pontos de função não-ajustados para a estimativa do projeto.

As 14 características estão detalhadas abaixo com suas respectivas tabelas de influencia:

1. Comunicação de Dados

A comunicação de dados analisa o nível em que a aplicação comunica-se diretamente com o processador. Os dados ou informações de controle utilizados na aplicação são enviados ou recebidos através de facilidades de comunicação.

Tabela 7 Característica de Comunicação de Dados

Grau	Descrição
0	Aplicação puramente batch ou funciona em um microcomputador <i>stand-alone</i> .
1	Aplicação é <i>batch</i> , mas possui entrada de dados remota ou impressão remota.
2	Aplicação é <i>batch</i> , mas utiliza entrada de dados remota e impressão remota.
3	Aplicação inclui coleções de dados online ou é um <i>front-end</i> para a execução de processamento batch ou de sistema de consulta.

4	Aplicação é mais do que um <i>front-end</i> , mas suporta apenas um tipo de protocolo de comunicação.
5	Aplicação é mais do que um <i>front-end</i> e suporta mais de um tipo de protocolo de comunicação.

(MANUAL IFPUG, 2005, p. 100)

Segundo a IFPUG (2005) normalmente a comunicação de dados para:

- Aplicações batch pontuam de 0 a 3.
- Aplicações on-line pontuam 4.
- Aplicações Web pontuam 4 ou 5.
- Sistemas real-time, de telecomunicações ou de controle de processos pontuam 4 ou 5.

2. Processamento Distribuído

O processamento distribuído representa o nível da transferência de dados entre os componentes da aplicação, levando em consideração apenas o escopo interno da fronteira da aplicação.

Tabela 8 Característica de Processamento Distribuído

Grau	Descrição
0	A aplicação não auxilia a transferência de dados ou o processamento de funções entre os componentes da aplicação.
1	A aplicação prepara dados para o usuário processar em outro componente do sistema, tal como planilhas em PC ou DBMS.
2	Dados são preparados para transferência, então são transferidos e processados em um outro componente da aplicação (não devem ser considerados como componentes nesse caso, processamentos feitos pelo usuário final).
3	Existe processamento distribuído e transferência de dados on-line apenas em uma direção.

4	Existe processamento distribuído e transferência de dados on-line em ambas as direções.
5	As funções de processamento são dinamicamente executadas no componente mais adequado do sistema.

(MANUAL IFPUG, 2005, p. 103)

Segundo IFPUG (2005) tipicamente a pontuação para o processamento distribuído é a seguinte:

- A maioria das aplicações, incluindo aplicações legadas, recebem 0.
- As aplicações distribuídas primitivas, inclusive aplicações batch em que dados não são transferidos online pontuam 1 ou 2.
- Aplicações cliente-servidor ou web recebem 3 ou 4.
- É raro uma nota 5.
- Existindo múltiplos servidores ou processadores, cada qual seria selecionado dinamicamente de acordo com sua disponibilidade para receber nota 5.

3. Performance

A performance analisa o tempo de resposta e a performance de transferência de dados no desenvolvimento da aplicação.

Os objetivos de performance da aplicação, solicitados e aprovados pelo usuário, tanto em tempo de resposta quanto em transferência de dados, influenciam (ou vão influenciar) o projeto, o desenvolvimento, a instalação e o suporte que será fornecido na aplicação.

Tabela 9 Característica de Performance

Grau	Descrição
0	Nenhum requerimento especial de performance foi solicitado pelo usuário.
1	Requerimentos de performance e de desenho foram estabelecidos e revistos, mas nenhuma ação especial foi requerida.

2	Tempo de resposta e volume de processamento são críticos durante horários de pico de processamento. Nenhuma determinação especial para a utilização do processador foi estabelecida. A data limite para a disponibilidade de processamento é sempre o próximo dia útil.
3	Tempo de resposta e volume de processamento são críticos durante todo o horário comercial. Nenhuma determinação especial para a utilização do processador foi estabelecida. A data limite para a comunicação com outros sistemas é um item importante.
4	Além de 3, os requerimentos de performance estabelecidos requerem tarefas de análise de performance na fase de análise e de desenho da aplicação.
5	Além de 4, ferramentas de análise de performance foram usadas nas fases de desenho, desenvolvimento e/ou implementação para atingir os requerimentos de performance estabelecidos.

(MANUAL IFPUG, 2005, p. 106)

Segundo a IFPUG (2005) tipicamente a performance segue as seguintes pontuações:

- Aplicações batch recebem nota de 0 a 4.
- Aplicações on-line (incluindo cliente-servidor interativo ou web) recebem de 0 a 4.
- Aplicações web recebem 4 ou 5.
- A maioria dos sistemas on-line MIS (Management Information System - sistema de informação gerencial recebe 2.
- Sistemas real-time, de telecomunicações ou controle de processos recebem de 0 a 5.
- Uma nota 5 requer o uso de ferramentas de análise de performance.

4. Configuração Altamente Utilizada

A configuração altamente utilizada descreve o quanto as restrições dos recursos computacionais influenciam no desenvolvimento da aplicação. Esta característica aparece quando existem algumas restrições operacionais por causa do uso pesado dos equipamentos.

Tabela 10 Característica de Configuração Altamente Utilizada

Grau	Descrição
0	Nenhuma restrição operacional explícita ou mesmo implícita foi incluída.
1	Existem restrições operacionais leves. Não é necessário esforço especial para resolver as restrições.
2	Algumas considerações de ajuste de performance e segurança são necessárias.
3	Requerimentos específicos quanto ao processador são necessários para uma parte definida da aplicação.
4	Restrições operacionais especificadas requerem cuidados especiais para a aplicação quando executando no processador central ou no processador dedicado.
5	Além do item 4, existem restrições especiais na distribuição dos componentes da aplicação no sistema.

(MANUAL IFPUG, 2005, p. 108)

Segundo a IFPUG (2005) a pontuação típica para a configuração altamente utilizada é a seguinte:

- A maioria das aplicações recebe nota 2.
- Cliente-servidor, web, real-time, telecomunicações ou sistemas de controle de processos recebem de 3 a 5, mas então você precisaria de um processador dedicado, ou de múltiplos processadores processando as mesmas transações e buscando os recursos mais eficientes de processamento.

5. Volume de Transações

O volume de transações diz respeito ao nível em que o volume de transações influencia no desenvolvimento da aplicação. Essa característica é bastante parecido com a performance, pois deve ser analisado em qual frequência e o quanto irá interferir no projeto, seja no desenvolvimento, na instalação ou no suporte oferecido na aplicação.

Tabela 11 Característica de Volume de Transações

Grau	Descrição
0	Não estão previstos períodos de picos de volume de transação.
1	Estão previstos picos de transações mensalmente, trimestralmente, anualmente ou em certo período do ano.
2	São previstos picos semanais.
3	São previstos picos diários.
4	Altos volumes de transações foram estabelecidos pelo usuário como requisitos da aplicação, ou o tempo de resposta necessário atinge nível alto o suficiente para requerer tarefas de análise de performance na fase de projeto.
5	Além do descrito no item anterior, é necessário utilizar ferramentas de análise de performance nas fases de desenho, desenvolvimento e/ou implantação.

(MANUAL IFPUG, 2005, p. 110)

Segundo a IFPUG (2005) a pontuação tipicamente utilizada para o volume de transações está listada a seguir:

- Aplicações batch recebem de 0 a 3.
- Aplicações on-line (incluindo interações de Cliente-servidor ou Web) recebem de 0 a 4.
- Sistemas real-time, de telecomunicações ou controle de processos recebem de 0 a 5.
- Uma nota 5 requer a utilização de ferramentas de análise de performance.

6. Entrada de Dados On-line

A entrada de dados on-line descreve em que nível os dados são fornecidos a aplicação através de transações interativas, ou seja, em tempo real.

Tabela 12 Característica de Entrada de Dados On-line

Grau	Descrição
0	Todas as transações são processadas em modo batch.
1	De 1% a 7% das transações são entradas de dados online.

2	De 8% a 15% das transações são entradas de dados online.
3	De 16% a 23% das transações são entradas de dados online.
4	De 24% a 30% das transações são entradas de dados online.
5	Mais de 30% das transações são entradas de dados online.

(MANUAL IFPUG, 2005, p. 112)

Segundo a IFPUG (2005) normalmente a pontuação usada é a seguinte:

- Aplicações batch recebem 0 ou 1.
- Aplicações on-line, real-time, de telecomunicações ou sistemas de controle de processos recebem 5.
- A maioria das aplicações on-line atuais (incluindo cliente-servidor interativo ou web) recebem 5.
- Sistemas batch com características on-line podem ter a maioria das transações batch, mas o sistema deve ser pelo menos 71 % batch para receber menos do que 5.

7. Eficiência do Usuário Final

A eficiência do usuário final analisa a eficiência do usuário final em relação à aplicação a ser medida, ou seja, é levado em consideração o nível de dificuldade oferecido as pessoas que utilizarão a aplicação.

Abaixo estão alguns recursos que devem ser considerados:

- Auxílio à navegação (teclas de atalho, acesso direto, menus dinâmico).
- Menus.
- Documentação e auxílio online.
- Movimentação automática de cursor.
- Scrolling (Movimentação de tela vertical e/ou horizontal através de setas).
- Impressão remota (através de transações on-line).
- Teclas de função preestabelecidas.
- Processos batch submetidos a partir de transações on-line.

- Seleção de cursor em campos da tela.
- Utilização intensa de campos com vídeo reverso, intensificados, sublinhados, coloridos e outros indicadores.
- Impressão da documentação das transações online.
- Utilização de mouse.
- Menus pop-up.
- O menor número possível de telas para executar o negócio.
- Suporte bilíngüe (suporte para duas linguagens – contar como 4 itens).
- Suporte multilíngüe (suporte para duas linguagens – contar como mais de 6 itens).

Tabela 13 Característica de Eficiência do Usuário Final

Grau	Descrição
0	Nenhum dos itens descritos acima.
1	De 1 a 3 itens acima.
2	De 4 a 5 itens acima.
3	Mais de 5 dos itens acima. Mas, não há requerimentos específicos do usuário quanto à amigabilidade do sistema.
4	Mais de 5 dos itens acima. Além disso, foram estabelecidos requerimentos quanto à amigabilidade fortes o suficiente para gerarem atividades na fase de projeto específicas envolvendo fatores como minimização da digitação, maximização de defaults e uso de templates.
5	Mais de 5 dos itens acima. Além disso, foram estabelecidos requerimentos quanto à amigabilidade fortes para requerer ferramentas e processos especiais para demonstrar que os objetivos foram alcançados.

(MANUAL IFPUG, 2005, p.114)

Segundo a IFPUG (2005) a pontuação usada normalmente nos projetos é a seguinte:

- Aplicações puramente batch recebem 0.
- Interface com o usuário em modo character recebe 1 ou talvez 2.
- Interface GUI para ser usada com baixo volume de transações recebe 3.

- Interface GUI para ser usada com alto volume de transações, assim como a
- maioria das interfaces de Intranet recebem 4 (devem existir tarefas de
- “design” referentes a fatores humanos).
- Interface com o usuário de Intranet recebe 5 (requer o uso de ferramentas e
- processos especiais para demonstrar que os objetivos foram alcançados).

8. Atualização On-Line

A atualização on-line verifica o nível em que os ALIs da aplicação são atualizados de forma on-line.

Tabela 14 Característica de Atualização On-Line

Grau	Descrição
0	Nenhum arquivo é atualizado online.
1	Atualização online de um a 3 arquivos de controle. O volume de atualizações é baixo e a recuperação de dados é fácil.
2	Atualização online 4 ou mais arquivos de controle. O volume de atualizações é baixo e a recuperação de dados é fácil.
3	Atualização online da maioria dos Arquivos Lógicos Internos.
4	Além do item 3, é necessária proteção contra perda de dados que foi desenhada e implementada no sistema.
5	Além do item 4, altos volumes trazem considerações de custo no processo de recuperação. Processos para automatizar a recuperação foram incluídos minimizando a intervenção do operador.

(MANUAL IFPUG, 2005, p. 115)

Segundo a IFPUG normalmente a pontuação atribuída para atualização on-line é a seguinte:

- As aplicações puramente batch recebem 0.
- Atualizações on-line de arquivos que modificam a forma segundo
- aplicação processa ou valida dados recebidos recebem 1 ou a atualização on-line dos dados persistentes do usuário recebe Aplicações MIS (Sistema de Informação Gerencial) recebem a maioria das aplicações GUI (Interface

Gráfica do Usuário) mais.

- Aplicações que utilizam recuperação programada como por roll back” ou “commit” recebem 4. Backup operacional ou considerado proteção contra perda de dados.
- Aplicações que executam recuperação de dados, reinicialização funções autocontidas em caso de erro do sistema recebem 5 pode requerer um operador para pressionar “enter” ou executar função mínima para iniciar o processo.

9. Processamento Complexo

O processamento complexo analisa o nível da lógica de processamento e o quanto influencia no desenvolvimento da aplicação.

Abaixo estão alguns itens que servem para ajudar na determinação do processamento complexo:

- Controle sensetivo (processamento especial de auditoria) e/ou processamento de segurança especifica da aplicação.
- Processamento lógico extensivo.
- Processamento matemático extensivo.
- Processamento gerando muitas exceções, resultando em transações incompletas que devem ser processadas novamente.
- Exemplo: Transações incompletas em terminais ATM causados por interrupções de teleprocessamento ou falhas nas validações
- Processamento complexo para manusear múltiplas possibilidades de entrada/saída
- Exemplo: Multimídia ou diferentes dispositivos independentes.

Tabela 15 Característica de Processamento Complexo

Grau	Descrição
0	Nenhum dos itens descritos acima.
1	Apenas 1 dos itens acima.

2	2 dos itens acima.
3	3 dos itens acima.
4	4 dos itens acima.
5	Todos itens acima.

(MANUAL IFPUG, 2005, p. 116)

Segundo a IFPUG (2005) esta pontuação independe da plataforma utilizada.

10. Reusabilidade

A reusabilidade refere-se quanto o código-fonte ou componente de uma aplicação, foram desenvolvidos, especificados e se oferecem suporte para serem reutilizados em outro sistema.

Tabela 16 Característica de Reusabilidade

Grau	Descrição
0	Nenhum código reutilizável.
1	Código reutilizável é utilizado somente na aplicação.
2	Menos de 10% de reutilização de código.
3	10% ou mais da aplicação prevê reutilização de código.
4	A aplicação foi especificamente gerada em pacotes (componentes) e/ou documentada para ter seu código facilmente reutilizado por outra aplicação e a aplicação é customizada pelo usuário no código fonte.
5	Aplicação especificamente projetada e/ou documentada para ter seu código facilmente reutilizado por outra aplicação e a aplicação pode ser customizada através de parâmetros que podem ser alterados pelo usuário.

(MANUAL IFPUG, 2005, p. 118)

Segundo a IFPUG (2005) esta pontuação independe da plataforma usada.

11. Facilidade de Instalação

A facilidade de instalação leva em consideração o esforço necessário para converter os ambientes que influenciam no desenvolvimento da aplicação, podendo ser uma conversão de caráter físico (hardware) ou lógico (software) no ambiente

para atender as necessidades da aplicação.

Tabela 17 Característica de Facilidade de Instalação

Grau	Descrição
0	Nenhuma consideração especial foi estabelecida pelo usuário e nenhum procedimento especial para configuração de hardware ou de software é necessário para implantação.
1	Nenhuma consideração especial foi estabelecida pelo usuário, mas procedimento especial para configuração de hardware ou de software é necessário para implantação.
2	Requerimentos de conversão e implantação foram estabelecidos pelo usuário e roteiros de conversão e implantação foram providos e testados. O impacto da conversão no projeto não é considerado importante.
3	Requerimentos de conversão e implantação foram estabelecidos pelo usuário e roteiros de conversão e implantação foram providos e testados. O impacto da conversão no projeto é 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.

(MANUAL IFPUG, 2005, p. 119)

Segundo a IFPUG (2005) é independente da plataforma a pontuação desta característica.

12. Facilidade de Operação

A facilidade de operação leva em consideração o nível de atenção que a aplicação dar a aspectos operacionais como processos de inicialização, de backup, e recuperação. Mas isso sem a intervenção manual para realização dessas tarefas.

Tabela 18 Característica de Facilidade Operação

Grau	Descrição
0	Nenhuma consideração especial de operação, além do processo normal de salva foi estabelecido pelo usuário.

1-4	<p>Verifique quais das seguintes afirmativas podem ser identificadas na aplicação. Selecione as que forem aplicadas. Cada item vale um ponto, exceto se definido explicitamente.</p> <ul style="list-style-type: none"> • Foram desenvolvidos processos de inicialização, salva e recuperação, mas a intervenção do operador é necessária. • Foram estabelecidos processos de inicialização, salva e recuperação, e nenhuma intervenção do operador é necessária (contar como 2 itens). • A aplicação minimiza a necessidade de montar fitas magnéticas. • A aplicação minimiza a necessidade de manuseio de papel.
5	<p>A aplicação foi desenhada para trabalhar sem operador; nenhuma intervenção do operador é necessária para operar o sistema além de executar e encerrar a aplicação. A aplicação possui rotinas automáticas para recuperação em caso de erro.</p>

(MANUAL IFPUG, 2005, p. 121)

Para a IFPUG (2005) a pontuação desta característica independe da plataforma.

13. Múltiplos Locais

Esta característica analisa se a aplicação foi projetada, desenvolvida e suportada para ser instalada em múltiplos locais para múltiplas organizações. Deve ser considerado se a aplicação é capaz de rodar em ambientes heterogêneos de software e hardware, ou apenas em ambientes similares.

Tabela 19 Característica de Múltiplos Locais

Grau	Descrição
0	Os requerimentos do usuário não consideram a necessidade de instalação em mais de um local.
1	A necessidade de múltiplos locais foi considerada no projeto, e a aplicação foi desenhada para operar apenas sobre idênticos ambientes de software e hardware.
2	A necessidade de múltiplos locais foi considerada no projeto, e a aplicação esta preparada para trabalhar apenas em ambientes similares de software e/ou hardware.
3	A necessidade de múltiplos locais foi considerada no projeto, e a aplicação esta preparada para trabalhar em diferentes ambientes de software e/ou hardware.
4	Plano de documentação e manutenção foram providos e testados para suportar a aplicação 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 a aplicação em múltiplos locais; além disso, o item 3 caracteriza a aplicação.
---	---

(MANUAL IFPUG, 2005, p. 123)

Já para esta característica a pontuação depende do número de plataformas diferentes.

14. Modificação Facilitada

Modificação facilitada como o próprio nome sugere é a facilidade que realizar mudanças na aplicação seja ela de lógica de processamento ou da estrutura de dados.

Abaixo estão algumas características que devem ser levadas em consideração:

- Consultas e relatórios flexíveis para atender necessidades simples (contar como 1 item).
- Consultas e relatórios flexíveis para atender necessidades de complexidade média (contar como 2 itens).
- Consultas e relatórios flexíveis para atender necessidades complexas (contar como 3 itens).
- Dados de controle são armazenados em tabelas que são mantidas pelo usuário através de processos on-line, mas mudanças são tornadas efetivas apenas no dia seguinte.
- Dados de controle são armazenados em tabelas que são mantidas pelo usuário através de processos on-line, as mudanças têm efeito imediato (contar como 2 itens).

Tabela 20 Característica de Modificação Facilitada

Grau	Descrição
0	Nenhum dos itens descritos acima.
1	Apenas 1 dos itens acima.
2	2 dos itens acima.
3	3 dos itens acima.

4	4 dos itens acima.
5	Todos itens acima.

(MANUAL IFPUG, 2005, p. 125)

Segundo a IFPUG (2005) esta característica independe da plataforma.

3.3.6. CÁLCULO DOS PONTOS DE FUNÇÃO AJUSTADOS

O Cálculo dos Pontos de função ajustados vai depender da finalidade da contagem, podendo ser de três tipos:

- Projeto de desenvolvimento
- Projeto de melhoria
- Aplicação

3.3.6.1. PROJETO DE DESENVOLVIMENTO

Para o cálculo de um projeto de desenvolvimento utiliza-se a seguinte fórmula:

DFP = (UFP + CFP) x VAF, onde DFP é o número de pontos de função do projeto de desenvolvimento; UFP é o número de pontos de função não-ajustados das funções disponíveis após a instalação; CFP é o número de pontos de função não-ajustados das funções de conversão; VAF: Valor do fator de ajuste.

3.3.6.2. PROJETO DE MELHORIA

Partindo do princípio adotado pelo IFPUG onde melhoria, diz respeito apenas a manutenções evolutivas do sistema, necessária para atender novas funcionalidades solicitadas pelo usuário, assim manutenções do tipo corretiva e preventivas não fazem parte.

A fórmula utilizada é a seguinte:

EFP = [(ADD + CHGA + CFP) x VAFA] + (DEL x VAFB), onde EFP é o número de pontos de função do projeto de melhoria; ADD é o número de pontos de função não-ajustados das funções incluídas pelo projeto de melhoria; CHGA é o

número de pontos de função não-ajustados das funções modificadas; CFP é o número de pontos não-ajustados adicionados pela conversão; VAFA é o valor do fator de ajuste da aplicação depois do projeto de melhoria; DEL é o número de pontos de função não-ajustados das funções excluídas pelo projeto de melhoria; VAFB é o valor do fator de ajuste da aplicação antes do projeto de melhoria.

3.3.6.3. APLICAÇÃO

Existem duas fórmulas para calcular o número de pontos de função da aplicação. Uma para a primeira contagem dos pontos de função da aplicação e outra para recalculá-lo seu tamanho após um projeto de melhoria ter alterado sua funcionalidade.

Na fórmula usada para Contagem Inicial representa todas as funcionalidades solicitadas pelo usuário de uma aplicação instalada, podendo ser um pacote de software, um software desenvolvido e instalado.

Formula para Contagem Inicial:

AFP = ADD x VAF, onde AFP é o número de pontos de função ajustados da aplicação; ADD é o número de pontos de função não-ajustados das funções instaladas; VAF é o valor do fator de ajuste da aplicação.

A fórmula utilizada para um projeto após Projeto de Melhoria, quando um projeto de melhoria é concluído, o número de pontos de função da aplicação deve ser atualizado para refletir as modificações na aplicação. A funcionalidade da aplicação pode ser aumentada através de uma nova funcionalidade adicionada a pedido do usuário; uma funcionalidade modificada pode aumentar diminuir ou não afetar o tamanho da aplicação; uma funcionalidade excluída diminui o tamanho da aplicação, entre outras.

Formula após projeto de melhoria:

APF = [(UFPB + ADD + CHGA) – (CHGB + DEL)] x VAFA, onde AFP é o número de pontos de função ajustados da aplicação; UFPB são os pontos de função não-ajustados da aplicação antes de melhoria; ADD são pontos de função não-ajustados das funções incluídas pelo projeto de melhoria; CHGA são pontos de função não-

ajustados das funções alteradas pelo projeto de melhoria depois de seu término; CHGB são pontos de função não-ajustados das funções alteradas pelo projeto de melhoria antes do seu término; DEL são pontos de função não-ajustados das funções excluídas pelo projeto de melhoria; VAFA é o valor do fator de ajuste depois do projeto de melhoria.

4. PROTÓTIPO DA FERRAMENTA

Como resultado do trabalho foi desenvolvido um protótipo de uma ferramenta de estimativa denominada de **APF**. A ferramenta tem o intuito de auxiliar os profissionais de TI na realização da estimativa de sistemas a partir da especificação de requisitos, baseando na análise de pontos por função e no tamanho da aplicação.

A idéia de desenvolver uma ferramenta desse tipo, deu-se pela necessidade das pequenas empresas de ter um software de estimativa fácil de ser utilizado, substituindo as planilhas muitas vezes enormes e confusas e possibilitando pessoas que não possuam muita experiência utilizem a técnica de PF e realizem uma estimativa com o auxílio da base histórica de outros projetos da empresa.

Ligado a isso oferecer um controle sobre a evolução e crescimento do tamanho da aplicação através de um controle de versões.

Assim visamos auxiliar as empresas na realização de estimativas precisas, pois é de extrema importância que a estimativa obtenha resultados o mais próximo da realidade.

4.1. FUNCIONALIDADES

A **APF** objetiva possuir um layout simples e intuitivo, na tentativa de melhorar a usabilidade. Baseia-se no fluxo de contagem de ponto de função, sendo que o controle dos projetos é feito através de versões do mesmo, a partir do tipo de contagem, garantindo assim um controle mais preciso. Tem integração com um banco de dados para permitir armazenamento do histórico das contagens realizadas, além de uma retro-alimentação, dados de outros projetos podem servir de entrada para os novos projetos. Permite ao usuário além do cadastro de novos projetos, a importação de projetos já existentes, sejam acrescentados sem muito esforço, aumentando a sua base histórica. Na parte de fatores de ajuste oferece

como melhoria a possibilidade de criar modelos de sistema (ex. sistemas web, utilizando a linguagem JAVA) aonde esses fatores virão com uma sugestão de valor, garantindo assim um ganho de conhecimento para os usuários inexperientes em APF.

Permitindo também a geração de relatórios comparativos entre versões do projeto e entre projetos distintos, gerando um indicador entre as diversas etapas do projeto (versões no sistema).

Além da contagem dos pontos de função a ferramenta fornece a possibilidade de cadastro de mão-de-obra, garantindo além da contagem de pontos uma estimativa de custos e tempo total, através do cruzamento dessas informações.

A estrutura de funcionalidades do sistema segue o fluxo do processo de análise de PF conforme descrição:

- Função de Acesso: Cadastro, exclusão, edição, consulta dos usuários que participam do gerenciamento e cadastro de dados na ferramenta, gerenciando permissões aos módulos do sistema.
- Manter Projetos: Cadastro, exclusão, edição, consulta dos projetos, e dados relacionados a eles (Nome, tipo de projeto, etc.), além de permitir um gerenciamento das versões do projeto.
- Manter Grupo de dados: Permite o cadastro, exclusão, edição, consulta dos grupos de dados e processos elementares, permite a importação desses dados de outros projetos previamente cadastrados.
- Manter Fatores de Ajustes: Módulo de seleção dos fatores de ajustes e o impacto dos mesmos, para um projeto selecionado. Permitindo a criação de modelos de sugestão dessas características de acordo com o tipo de projeto.
- Cálculo dos Pontos por Função: Módulo responsável pelo cálculo do PF não-ajustados e os PF ajustados.
- Relatórios e Gráficos: Geração de relatório e gráficos comparativos entre os projetos cadastrados e suas versões.
- Manter Mão-de-obra: Permite o cadastro, exclusão, edição, consulta de mão-de-obra, para cálculo e estimativa de tempo do projeto.

4.2. A ARQUITETURA DO APF

O padrão de arquitetura de software adotado será o Model-View-Controller (MVC), pois com este modelo as camadas de dados (model), layout (view) e de controle (controler), ficam independentes, onde alterações realizadas em uma camada não causarão impacto nas outras camadas.

Na camada de layout (view) as páginas *web* foram criadas como JSP, nesta camada utilizamos o framework Struts versão 1.3, e nela é realizada a integração com a camada de controle. Na camada de controle é onde se encontram as classes das funcionalidades do sistema, é nesta camada que ficam as regras de negócio, nela se encontram as ações como cadastramento, edição, listagem, entre outros.

A camada de controle interage com a camada de dados (model), sendo esta responsável pela comunicação com o banco de dados. No seu desenvolvimento usamos o framework de persistência Hibernate.

No ambiente de desenvolvimento foi utilizado computadores com sistemas operacionais Windows XP, ferramentas para a construção do sistema destacamos o Netbeans 6.0.1, para modelagem dos diagramas UML foi utilizado o Jude Community 5.0, para os relatórios o iReports 2.0.5, o banco de dados MySQL 5.0 para armazenamento das funcionalidades e características da técnica de ponto de função, dos dados referentes às análises dos projetos cadastrados pelos usuários e os próprios dados dos usuários. Para servidor de aplicações integrando ao *webserver* utilizamos o *Tomcat*³ versão 5.5 como servidor de aplicações integrado ao *webserver*, que suporta os *Java Servlets*⁴ e as páginas *JSP*⁵, usados na implementação do sistema.

³ O Tomcat é um servidor de aplicações Java para *web*, distribuído como software livre, para as tecnologias Java Servlet e JavaServer Pages (JSP).

⁴ Servlet é uma tecnologia utilizada para o desenvolvimento de aplicações Web com conteúdo dinâmico. Ele contém uma API que abstrai e disponibiliza os recursos do servidor Web de maneira simplificada para o programador.

⁵ JSP (Java Server Pages) é uma tecnologia utilizada no desenvolvimento de aplicações para *web*. Por ser baseada na linguagem de programação Java as JSP se tornam portáteis e independente de plataforma.

4.2.1. CAMADA DE MODELO

Na camada de modelo do sistema é onde se encontra todas as regras de negócio do sistema. Nesta camada existe uma pacote contendo todas as classes responsáveis pelo acesso ao banco de dados e como utilizar os dados recuperados.

Esta camada também é responsável pela persistência de dados no banco de dados e pela comunicação com o mesmo.

No APF podemos citar algumas das principais classes para essa camada e sua respectiva função:

- i. A classe DadosBO realiza o controle das tarefas relacionadas aos grupos de dados e processos elementares, como validação dos campos, importação e complexidade.
- ii. A classe Calculo é responsável por implementar os métodos de calculo de pontos de função bruto, complexidade e ponto de função ajustados
- iii. A classe VersaoBO realiza o controle das operações sobre a versão do projeto, como por exemplo o cálculo ou atualização dos pontos de função brutos, quando é adicionada uma funcionalidade ao cadastro.

4.2.1.1. DESCRIÇÃO DAS CLASSES E SUAS ASSOCIAÇÕES

Entre as principais classe definidas no projeto podemos destacar as descritas abaixo:

- **Projeto:** Representa um projeto.
- **Versão:** Representa uma versão do projeto.
- **Dados:** Representa cada função do projeto (Processos Elementares e Grupo de dados).
- **Características:** Representa as características do sistema e sua resposta.
- **Resposta:** Representa as possíveis respostas para as características.
- **Modelo:** Representa um modelo de respostas para características.
- **Usuário:** Representa um usuário que tem acesso ao sistema.

- **Função:** Representa cada função do sistema
- **Mão-de-obra:** Representa mão-de-obra a ser alocada para o projeto.

4.2.1.2. PERSISTÊNCIA DE DADOS

Persistência de dados referencia-se a uma ação que consiste em manter os dados em meio físico recuperável, como banco de dados, de modo a garantir a permanência das informações de um determinado estado de um objeto lógico. Assim podem-se recuperar os dados mesmo depois da finalização do programa.

Na ferramenta APF foi utilizado o *framework Hibernate* para realizar a persistência dos dados no banco de dados *Mysql*. Para isto é necessário realizar o mapeamento do banco de dados no framework.

Com a utilização do Hibernate a complexidade diminui dos programas Java, pois ele transforma as classes em Java para tabelas de dados, incluindo os tipos de dados Java para os SQL. Tornando assim a aplicação portátil, porém gerando uma pequena perda de performance, devido a esta conversão.

As consultas foram feitas utilizando HQL (*Hibernate Query Language*) que é uma extensão da linguagem SQL (*Structured Query Language*), sendo possível a utilização do SQL. Desta forma reduzindo a complexidade para o desenvolvimento de um projeto OO (Orientado a Objeto), pois o desenvolvedor utiliza apenas as classes persistentes que por sua vez relaciona-se com o banco de dados.

Para configuração do Hibernate usamos o arquivo *hibernate.cfg.xml*, onde fica armazenada as configurações de conexão com o banco de dados e o mapeamento dos bancos.

Foi utilizada uma classe auxiliar, o *HibernateUtilityDAO*, que tem como funcionalidade abrir e fechar sessões, sendo ele o responsável pelo controle de transações na aplicação.

Para cada entidade do banco de dados foi criado um arquivo XML, feito para o Hibernate, que é responsável pelo mapeamento dessas entidades.

4.2.1.3. BASE DE DADOS

No APF, foi utilizado o *MySql* 5.0 como banco de dados. Porém com a utilização do *Hibernate*, e com as consultas feitas em HQL, a aplicação torna-se independente do banco de dados.

4.2.1.4. PADRÃO DAO (DATA ACCESS OBJECT)

“DAO (Data Access Object) é um padrão para persistência de dados que permite separar regras de negócio das regras de acesso a banco de dados.” (Wikipedia, 2008, p.1)

No APF foi utilizado os DAOs, pois as funcionalidades relacionadas com o banco de dados como conexões, mapeamento dos objetos Java para os tipo de dados SQL, as exceções, os retornos e acesso aos métodos do Hibernate, são realizados pelas classes de DAO.

4.2.2. CAMADA DE CONTROLE

A camada de controle é responsável pela comunicação Model-View. É ela que processa e responde a eventos, geralmente ações do usuário, e pode invocar alterações no Model.

No APF, a camada foi dividida por ações relacionadas a entidades, onde cada entidade possui ao menos uma classe na camada de controle, ação que implementa ao menos as funcionalidades apresentadas abaixo:

- I. limpar() - Limpa os dados existente das consultas da sessão;
- II. editar() - Realiza o cadastro ou alteração de um registro;
- III. pesquisar() - Possibilita a pesquisa e exibição de registro;
- IV. obterPorId() - Obtém a informação de um registro, permitindo assim sua visualização e alteração;
- V. deletar() - Possibilita a exclusão de um registro.

4.2.3. CAMADA DE VISUALIZAÇÃO

A camada de visualização (View), refere-se à interface com o usuário. Ela foi desenvolvida utilizando páginas JSP (Java Server Pages⁶), responsável pela apresentação de dados e encaminhar as solicitações ao controlador.

Para utilização desse recurso foram utilizado o framework Struts, além de algumas funções em javascript para validação de dados. Foi utilizado também no desenvolvimento do sistema a biblioteca javascript, JQuery, facilitando a utilização do AJAX, que é responsável pelo uso sistemático de diversas tecnologias, cujo o objetivo é modificar partes da página JSP sem a necessidade da recarga completa da página.

4.3. ESTUDO DE CASO

Visando um maior entendimento da análise de ponto de função, realizamos a análise e modelagem do sistema, assim como a contagem dos pontos de função em momentos distintos, dados que citamos a seguir:

4.3.1. CASOS DE USO

Os seguintes casos de uso foram aferidos:

- Logar no sistema
- Manter Usuário
- Manter Projeto
- Controlar Versão do Projeto
- Comparar Projeto
- Manter Grupo de Dados
- Importar Grupo de Dados
- Alimentar Características Gerais

⁶ É uma tecnologia que permite ao programador produzir aplicações que acessem o banco de dados, manipulem arquivos no formato texto, capture informações a partir de formulários e capturem informações sobre o visitante e sobre o servidor.

- Manter Mão-de-obra
- Relacionar Mão-de-obra
- Manter Categoria
- Manter Modelo
- Calcular PF Bruto
- Calcular PF Ajustado
- Emitir relatório de comparação de projetos
- Emitir relatório de acompanhamento do projeto

O detalhamento desses casos de usos serão mostrados no Apêndice A deste trabalho.

4.3.2. CONTAGEM DE PONTOS DE FUNÇÃO DO ESTUDO DE CASO

Para garantir um acompanhamento do desenvolvimento do projeto, realizamos a estimativa do tamanho do software, através da contagem de ponto de função, que disponibilizamos abaixo:

Funcionalidades					
#	Descrição	Tipo	TD	AR/TR	Complexidade
1	Função	AIE	1	2	BAIXA
2	Resposta característica	AIE	1	3	BAIXA
3	Tipo característica	AIE	1	3	BAIXA
4	Usuário	ALI	2	8	BAIXA
5	Projeto	ALI	1	6	BAIXA
6	Categoria	ALI	1	3	BAIXA
7	Versão	ALI	1	5	BAIXA
8	Dados	ALI	2	12	BAIXA
9	Tipo	AIE	1	2	BAIXA
10	Características	ALI	1	4	BAIXA
11	Modelo	ALI	2	5	BAIXA
12	Mão-de-obra	ALI	2	6	BAIXA

13	Logar no sistema	SE	1	5	BAIXA
14	Consultar usuário	CE	1	7	BAIXA
15	Cadastrar usuário	EE	1	9	BAIXA
16	Alterar usuário	EE	1	9	BAIXA
17	Excluir usuário	EE	1	3	BAIXA
18	Consultar projeto	CE	1	7	BAIXA
19	Cadastrar projeto	EE	1	6	BAIXA
20	Alterar projeto	EE	1	6	BAIXA
21	Excluir projeto	EE	1	3	BAIXA
22	Criar versão do projeto	EE	2	3	BAIXA
23	Liberar versão do projeto	EE	1	2	BAIXA
24	Consultar dados	CE	2	7	MÉDIA
25	Cadastrar dados	EE	2	8	MÉDIA
26	Alterar dados	EE	2	8	MÉDIA
27	Excluir dados	EE	2	3	MÉDIA
28	Preencher características gerais	EE	3	7	ALTA
29	Calcular PF	SE	4	12	ALTA
30	Calcular PF ajustado	SE	6	12	ALTA
31	Relatório comparativo entre projetos	SE	4	16	ALTA
32	Relatório indicador do projeto	SE	4	16	ALTA
33	Cadastrar mão-de-obra	EE	1	6	BAIXA
34	Alterar mão-de-obra	EE	1	6	BAIXA
35	Consultar mão-de-obra	CE	1	9	BAIXA
36	Excluir mão-de-obra	EE	1	3	BAIXA
37	Relacionar mão-de-obra	EE	1	7	BAIXA
38	Selecionar projeto	CE	2	5	BAIXA
39	Importar dados	CE	3	7	MÉDIA
40	Cadastrar categoria	EE	1	4	BAIXA

41	Alterar categoria	EE	1	4	BAIXA
42	Consultar categoria	CE	1	4	BAIXA
43	Excluir categoria	EE	1	3	BAIXA
44	Cadastrar modelo	EE	3	9	ALTA
45	Alterar modelo	EE	3	9	ALTA
46	Consultar modelo	CE	1	7	BAIXA
47	Excluir modelo	EE	1	3	BAIXA
48	Usar modelo	CE	1	3	BAIXA
49	Calcular estimativa do projeto	SE	1	4	BAIXA

Legenda:
(Tipo) - Classificação da funcionalidade (ALI, AIE, EE, SE ou CE)
(TD) - Quantidade de tipos de dados
(AR) - Quantidade de arquivos referenciados
(TR) - Quantidade de tipos de registros

Cálculo dos ponto de função não-ajustados						
Tipo de Função		Complexidade funcional			Totais por complexidade	Totais por tipo de função
EE	16	BAIXA	x 3	48		
	<u>3</u>	MÉDIA	x 4	<u>12</u>		
	<u>3</u>	ALTA	x 6	<u>18</u>		78
	<u> </u>			<u> </u>		<u> </u>
SE	2	BAIXA	x 4	8		
	<u>0</u>	MÉDIA	x 5	<u>0</u>		
	<u>4</u>	ALTA	x 7	<u>28</u>		36
	<u> </u>			<u> </u>		<u> </u>

CE	7	BAIXA	x 3	21	
	<u>2</u>	MÉDIA	x 4	<u>8</u>	
	<u>0</u>	ALTA	x 6	<u>0</u>	29
	—			—	—
ALI	8	BAIXA	x 7	56	
	<u>0</u>	MÉDIA	x 10	<u>0</u>	
	<u>0</u>	ALTA	x 15	<u>0</u>	56
	—			—	—
AIE	4	BAIXA	x 5	20	
	<u>0</u>	MÉDIA	x 7	<u>0</u>	
	<u>0</u>	ALTA	x 10	<u>0</u>	20
	—			—	—
	48				
Total de pontos de função não-ajustados					<u>219</u>

Características Gerais de Sistema	NI
01 - Comunicação de Dados	4
02 - Processamento Distribuído	4
03 - Performance	0
04 - Configuração Altamente Utilizada	2
05 - Volume de Transações	0
06 - Entrada de Dados On-line	5
07 - Eficiência do Usuário Final	3
08 - Atualização On-Line	3
09 - Processamento Complexo	0

10 - Reusabilidade	2
11 - Facilidade de Instalação	2
12 - Facilidade de Operação	2
13 - Múltiplos Locais	3
14 - Modificação Facilitada	2
Total dos Níveis de Influência (TDI)	32
Valor do Fator de Ajuste (VAF)	0,97

Ponto de função não ajustados	215
Fator de Ajustes	0,97
Ponto de função da aplicação	212

4.3.3. DIAGRAMA DE ENTIDADE RELACIONAL (DER)

A partir dos Estudos de Caso foi definido o DER do sistema, representado pela figura a seguir:

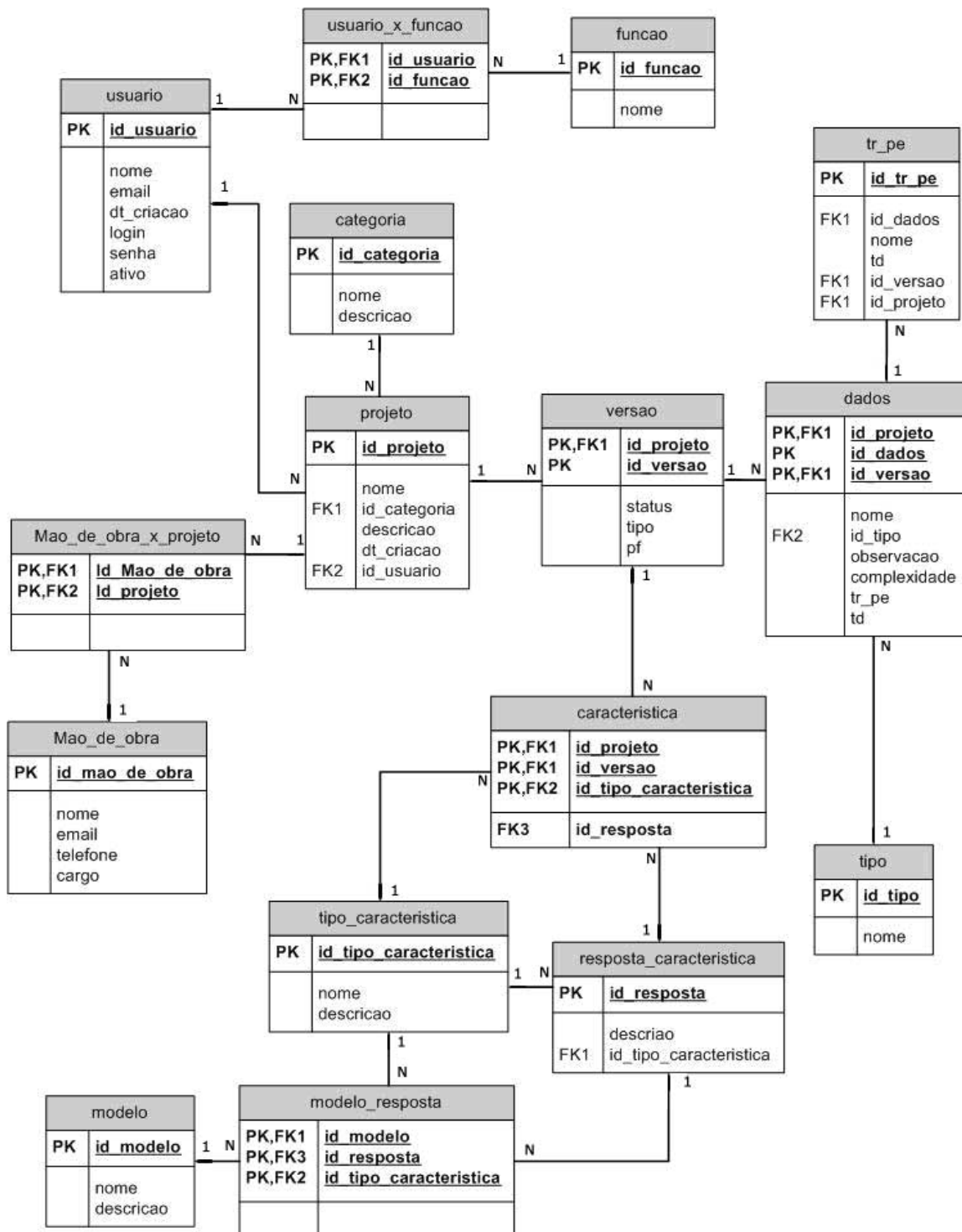


Figura 4 DER do Sistema APF

5. CONCLUSÃO

Este trabalho teve como objetivo demonstrar a técnica de análise de ponto de função, e por seguinte a construção da ferramenta APF de apoio a contagem de pontos de função, para o cálculo do tamanho da aplicação.

Foi mostrado também, através de conceitos de engenharia de software, métricas e suas medições, vantagens de sua utilização, e como servem de base para os cálculos de estimativas.

No contexto do desenvolvimento da ferramenta APF, foi apresentado um desvio de 30% entre o tempo calculado e o tempo realizado, valor que ao ser analisado deve levar em consideração a nossa falta de experiência em estimativas de projetos e que a técnica apresenta uma variação de -35% a 35%, Essa margem de erro foi alcançada utilizando-se o valor de três horas por ponto de função, tendo como base o valor utilizado pelas empresas observadas, que variam esse valor entre três e quatro horas.

Uma questão importante a ser destacada são os fatores de ajustes, pois os mesmos fornecem características específicas de um projeto, mas possuem um caráter subjetivo, ou seja, dependem da experiência do profissional em projetos parecidos com o que está sendo desenvolvido.

Dessa forma o IFPUG possui uma equipe focada na análise do fator de ajuste. Essa equipe está estudando uma atualização dessas características, que em alguns casos estão desatualizadas ou são semelhantes.

Com posse dessa informação modificamos a ferramenta, possibilitando o usuário cadastrar seus fatores para realização do ajuste ou até mesmo não utilizá-lo para o cálculo da estimativa.

A principal dificuldade desse trabalho foi encontrar projetos que servissem de base histórica tanto nos testes da ferramenta quanto para o auxílio da contagem da mesma. Para tanto foram utilizados 6 projetos, além da posterior utilização da contagem da própria ferramenta.

Como proposta de melhoria da ferramenta, poderia ser realizados estudos em outros tipos de métricas a serem adicionadas na ferramenta, aumentando assim a precisão da estimativa e possibilitando o usuário diversas visões do mesmo projeto. E possibilitar a importação de diagramas UML, automatizando mais o processo.

APÊNDICE A – CASOS DE USO

Este apêndice contém os casos de uso do APF que foram identificados, no entanto, não foram apresentados anteriormente.

Número: 1

Caso de uso: Logar no sistema.

Descrição: Liberar acesso as funcionalidades do sistema.

Atores: Usuário.

Pré-condição: Possuir Login e senha cadastrada no sistema.

Ações/Respostas:

Curso Normal:

- 1 - Ação - Usuário informa login e senha de acesso.
- 2 - Resposta - Sistema verifica se o usuário existe.
- 3 - Resposta - Se o usuário existir, o sistema verifica suas permissões de acesso.
- 4 - Resposta - Se o usuário tiver permissão, terá acesso às funcionalidades liberadas para o mesmo.

Curso Alternativo:

- 3 - Resposta - Se o usuário não existir, o sistema irá emitir mensagem: "Login Inválido".
- 4 - Resposta - Se o usuário não possuir permissão, o sistema deverá emitir mensagem: "Usuário sem permissão. Favor, entrar em contato com o administrador do sistema".

Número: 2

Caso de uso: Consultar usuário.

Descrição: Esta ação permite a consulta de usuários cadastrados na base de dados.

Atores: Administrador.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:**Curso Normal:**

1 - Ação - O usuário solicita consulta de usuários, podendo filtrá-lo pelo nome, e-mail ou login.

2 - Resposta - Se o usuário filtrar a consulta, o sistema deverá listar somente os usuários que atendam os filtros escolhidos e exibir as informações abaixo listadas:

- Nome
- Login
- E-mail
- Botão de cadastro
- Botão de alteração
- Botão de Exclusão

3 - Ação - O usuário seleciona o botão de cadastro de usuários.

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 3.

Curso Alternativo 1:

3 - Ação - O usuário seleciona o botão de alteração de usuários

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 4.

Curso Alternativo 2:

3 - Ação - O usuário seleciona o botão de exclusão de usuários.

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 5.

Curso Alternativo 3:

2 - Resposta - Se o usuário não filtrar a consulta, o sistema deverá exibir todos os usuários cadastrados.

Curso Alternativo 4:

2 - Resposta - Se não houver usuário cadastrado, exibir a mensagem ao usuário: "Nenhum usuário cadastrado" e disponibilizar o cadastro de usuário

seguindo o curso normal dos eventos descrito no caso de uso

Número: 3

Caso de uso: Cadastrar usuário.

Descrição: Esta ação permite o cadastro de usuário que utilizarão o sistema.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:**Curso Normal:**

1 - Ação - O usuário solicita cadastro de usuários.

2 - Resposta - O sistema constrói formulário com os seguintes dados:

- Nome
- E-mail
- Data da criação
- Login
- Senha
- Módulos do sistema
- Ativo

3 - Ação - O usuário informa os dados requeridos.

4 - Resposta - O sistema valida os dados informados.

5 - Resposta - Se os dados forem válidos, o cadastro será efetivado e a seguinte mensagem será exibida: "Usuário cadastrado com sucesso".

Curso Alternativo:

5 - Resposta - Se os dados não forem válidos, emitir a mensagem: "Campo X inválido".

Número: 4

Caso de uso: Alterar usuário.

Descrição: Permite atualização dos usuários que utilizarão o sistema.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário solicita atualização do usuário.

2 - Resposta - O sistema exibe o formulário com os dados descritos no curso normal de eventos do caso de uso 3, preenchidos.

3 - Ação - O usuário informa os dados requeridos.

4 - Resposta - O sistema valida os dados informados.

5 - Resposta - Se os dados forem válidos, o cadastro será efetivado e a seguinte mensagem será exibida: "Usuário cadastrado com sucesso".

- Curso Alternativo:

5 - Resposta - Se os dados não forem válidos, emitir a mensagem: "Campo X inválido".

Número: 5

Caso de uso: Excluir usuário.

Descrição: Permite a exclusão de usuários.

Atores: Administrador.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:

- Curso Normal:

1 - Ação - O administrador seleciona a exclusão do usuário.

2 - Resposta - O sistema verifica se há ligações de outros registros ao registro que será excluído. Se não houver, registra a (s) exclusão (ões), e exibe a mensagem: "Dados excluídos com sucesso".

- Curso Alternativo:

2 - Resposta - Se identificar que existem ligações com o registro que se deseja excluir, o sistema não efetiva a exclusão e deverá emitir a mensagem: "Registro não pôde ser removido, pois existem outros registros que fazem referência a ele".

Número: 6

Caso de uso: Consultar projeto.

Descrição: Esta ação permite a consulta de projetos cadastrados na base de dados.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário solicita consulta de projetos, podendo filtrá-lo pelo nome, descrição ou categoria.

2 - Resposta - Se o usuário filtrar a consulta, o sistema deverá listar somente os usuários que atendam os filtros escolhidos e exibir as informações abaixo listadas:

- Nome
- Descrição
- Categoria
- Botão de cadastro
- Botão de alteração
- Botão de Exclusão

3 - Ação - O usuário seleciona o botão de cadastro de projetos.

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 7.

- Curso Alternativo 1:

3 - Ação - O usuário seleciona o botão de alteração de projetos

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 8.

- Curso Alternativo 2:

3 - Ação - O usuário seleciona o botão de exclusão de projetos.

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 9.

- Curso Alternativo 3:

2 - Resposta - Se o usuário não filtrar a consulta, o sistema deverá exibir todos os projetos cadastrados.

- Curso Alternativo 4:

2 - Resposta - Se não houver projetos cadastrado, exibir a mensagem ao usuário: "Nenhum projetos cadastrado" e disponibilizar o cadastro de projetos seguindo o curso normal dos eventos descrito no caso de uso

Número: 7

Caso de uso: Cadastrar projeto.

Descrição: Esta ação permite o cadastro de projetos.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:**- Curso Normal:**

1 - Ação - O usuário solicita cadastro de projetos.

2 - Resposta - O sistema constrói formulário com os seguintes dados:

- Nome
- Descrição
- Categoria
- Data da criação

3 - Ação - O usuário informa os dados requeridos.

4 - Resposta - O sistema valida os dados informados.

5 - Resposta - Se os dados forem válidos, o cadastro será efetivado e a seguinte mensagem será exibida: "Projeto cadastrado com sucesso".

- Curso Alternativo:

5 - Resposta - Se os dados não forem válidos, emitir a mensagem: "Campo X inválido".

Número: 8

Caso de uso: Alterar projeto.

Descrição: Permite atualização dos projetos.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário solicita atualização do projeto.

2 - Resposta - O sistema exibe o formulário com os dados descritos no curso normal de eventos do caso de uso 7, preenchidos.

3 - Ação - O usuário informa os dados requeridos.

4 - Resposta - O sistema valida os dados informados.

5 - Resposta - Se os dados forem válidos, o cadastro será efetivado e a seguinte mensagem será exibida: "Projeto cadastrado com sucesso".

- Curso Alternativo:

5 - Resposta - Se os dados não forem válidos, emitir a mensagem: "Campo X inválido".

Número: 9

Caso de uso: Excluir projeto.

Descrição: Permite a exclusão de projetos.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário seleciona a exclusão do projeto.

2 - Resposta - O sistema verifica se há ligações de outros registros ao registro que será excluído. Se não houver, registra a (s) exclusão (ões), e exibe a mensagem: "Dados excluídos com sucesso".

- Curso Alternativo:

2 - Resposta - Se identificar que existem ligações com o registro que se

deseja excluir, o sistema não efetiva a exclusão e deverá emitir a mensagem: "Registro não pôde ser removido, pois existem outros registros que fazem referência a ele".

Número: 10

Caso de uso: Criar versão do projeto.

Descrição: Permite a criação de uma versão do projeto para edição das suas funções.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade, existir um projeto previamente cadastrado.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário seleciona a criação de uma nova versão do projeto, selecionando o tipo (Desenvolvimento, melhoria e aplicação).

2 - Resposta - O sistema verifica se não existem versões em edição.

3 - Resposta - O sistema verifica se existem versões prévias do Projeto.

4 - Resposta - Se for identificado versões anteriores todos os dados da última versão serão copiadas para essa nova versão, e deverá ser emitida a mensagem "Versão x.x liberada para edição".

- Curso Alternativo:

2 - Resposta - Se identificar que existe uma versão em edição, o sistema não efetiva a criação da nova versão e deverá emitir a mensagem: "Não foi possível criar uma nova versão, já existe uma versão em edição"

- Curso Alternativo:

3 - Resposta - Se identificar que não existem versões anteriores, deverá ser criado uma versão e emitido a mensagem "Versão 1.0 liberada para edição"

Número: 11

Caso de uso: Liberar versão do projeto.

Descrição: Permite a liberação de uma versão do projeto.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade, existir um projeto previamente cadastrado com uma versão em edição.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário seleciona a liberação da do projeto.

2 - Resposta - O sistema deverá emitir a mensagem "Após a liberação dessa versão os dados contidos na mesma não poderão ser modificados, deseja continuar?" e permitir ao usuário selecionar se deseja continuar com a liberação

3 - Ação - O usuário informa se deseja continuar com a liberação.

4 - Resposta - O status da versão é modificado para liberado e seus dados não podem mais ser modificados, deverá ser emitida a mensagem: "Versão liberada com sucesso".

- Curso Alternativo:

4 - Resposta - O sistema retorna para a tela anterior.

Número: 12

Caso de uso: Consultar Dados.

Descrição: Esta ação permite a consulta de Dados cadastrados na base de dados.

Atores: Usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade e existir uma versão em edição de um projeto.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário solicita consulta de dados, podendo filtrá-lo pelo nome, tipo ou complexidade.

2 - Resposta - Se o usuário filtrar a consulta, o sistema deverá listar somente os Dados que atendam os filtros escolhidos e exibir as informações abaixo listadas:

- Nome

- Tipo

- Complexidade
- Botão de cadastro
- Botão de alteração
- Botão de Exclusão

3 - Ação - O usuário seleciona o botão de cadastro de Dados.

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 13.

- Curso Alternativo 1:

3 - Ação - O usuário seleciona o botão de alteração de dados

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 14.

- Curso Alternativo 2:

3 - Ação - O usuário seleciona o botão de exclusão de Dados.

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 15.

- Curso Alternativo 3:

2 - Resposta - Se o usuário não filtrar a consulta, o sistema deverá exibir todos os Dados cadastrados.

- Curso Alternativo 4:

2 - Resposta - Se não houver Dados cadastrado, exibir a mensagem ao usuário: "Nenhum Dado cadastrado" e disponibilizar o cadastro de dados seguindo o curso normal dos eventos descrito no caso de uso

Número: 13

Caso de uso: Cadastrar Dados.

Descrição: Esta ação permite o cadastro de Dados.

Atores: Usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade e existir uma versão em edição de um projeto.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário solicita cadastro de Dados informado se realizará o cadastro simplificado ou detalhado.

2 - Resposta - O sistema constrói o formulário Simplificado com os seguintes dados:

- Nome
- Tipo
- Quantidade de Tipos de registro
- Quantidade de Tipo de dados
- Observação
- Complexidade

3 - Ação - O usuário informa os dados requeridos.

4 - Resposta - O sistema valida os dados informados.

5 - Resposta - Se os dados forem válidos, o cadastro será efetivado e a seguinte mensagem será exibida: "Dados cadastrado com sucesso".

- Curso Alternativo 1:

2 - Resposta - O sistema constrói o formulário Detalhado com os seguintes dados:

- Nome
- Tipo
- Observação
- Complexidade
- Botão Adicionar Tipo de Registro

3 - Ação - O usuário seleciona o Botão Adicionar Tipo de Registro.

4 - Resposta - O sistema adiciona ao formulário os seguintes dados:

- Nome Tipo de Registro.
- Quantidade de Tipo de dados.

5 - Ação - O usuário informa os dados requeridos.

6 - Resposta - O sistema valida os dados informados.

7 - Resposta - Se os dados forem válidos, o cadastro será efetivado e a

seguinte mensagem será exibida: “Dados cadastrado com sucesso”.

- Curso Alternativo 1:

5 - Resposta - O usuário seleciona o Botão Adicionar Tipo de Registro.

6 - Ação - O sistema seguirá o curso normal dos eventos descrito no Curso alternativo 1, item 4.

- Curso Alternativo 2:

7 - Resposta - Se os dados não forem válidos, emitir a mensagem: “Campo X inválido”.

Observação: O campo Complexidade é um campo calculado automaticamente.

Número: 14

Caso de uso: Alterar Dados.

Descrição: Permite atualização dos Dados.

Atores: Usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade e existir uma versão em edição de um projeto.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário solicita atualização de dados.

2 - Resposta - O sistema exibe o formulário com os dados descritos no curso normal de eventos do caso de uso 13, preenchidos.

3 - Ação - O usuário informa os dados requeridos.

4 - Resposta - O sistema valida os dados informados.

5 - Resposta - Se os dados forem válidos, o cadastro será efetivado e a seguinte mensagem será exibida: “Dados cadastrado com sucesso”.

- Curso Alternativo:

5 - Resposta - Se os dados não forem válidos, emitir a mensagem: “Campo X inválido”.

Número: 15

Caso de uso: Excluir Dados.

Descrição: Permite a exclusão de Dados.

Atores: Usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade e existir uma versão em edição de um projeto.

Ações/Respostas:

- **Curso Normal:**

1 - Ação - O administrador seleciona a exclusão dos Dados.

2 - Resposta - O sistema verifica se há ligações de outros registros ao registro que será excluído. Se não houver, registra a (s) exclusão (ões), e exibe a mensagem: "Dados excluídos com sucesso".

- **Curso Alternativo:**

2 - Resposta - Se identificar que existem ligações com o registro que se deseja excluir, o sistema não efetiva a exclusão e deverá emitir a mensagem: "Registro não pôde ser removido, pois existem outros registros que fazem referência a ele".

Número: 16

Caso de uso: Preencher Características Gerais

Descrição: Esta ação permite o acesso às respostas das características gerais.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade e existir uma versão em edição de um projeto.

Ações/Respostas:

- **Curso Normal:**

1 - Ação - O usuário solicita acesso ao preenchimento das características gerais.

2 - Resposta - O sistema preenche a lista com as características cadastrada gerais no banco.

3 - Ação - O usuário seleciona uma característica.

4 - Resposta - O sistema preenche a informação da pergunta daquela característica, carregando as opções de resposta existente no banco de acordo com a tabela abaixo.

5 - Ação - O usuário seleciona uma das respostas

6 - Resposta - O sistema armazena a opção selecionada.

Número: 17

Caso de uso: Calcular PF.

Descrição: Esta ação permite ao usuário o detalhamento dos ponto de função bruto do projeto.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade e existir uma versão em edição de um projeto.

Ações/Respostas:

- **Curso Normal:**

1 - Ação - O usuário solicita Calculo dos pontos de função do projeto.

2 - Resposta - O sistema lista o nome de todos os grupos de dados e processos elementares seguidos da complexidade e a quantidade de pontos de função de cada um deles de acordo com a tabela abaixo, além do total do somatório de todos os pontos de funções apresentados.

Tabela 21 Contribuição para Contagem de PF ((VASQUEZ; SIMÕES e ALBERT, 2003, p. 85 e 99)

Tipo de Função	Baixa	Média	Alta
Arquivo Lógico Interno	7 PF	10 PF	15 PF
Arquivo de Interface Externa	5 PF	7 PF	10 PF

Entrada Externa	3 PF	4 PF	6 PF
Saída Externa	4 PF	5 PF	7 PF
Consulta Externa	3 PF	4 PF	6 PF

Número: 18

Caso de uso: Calcular PF Ajustado

Descrição: ação permite ao usuário o detalhamento dos ponto de função ajustado

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade e existir uma versão em edição de um projeto.

Ações/Respostas:

1 - Ação - O usuário solicita Calculo dos pontos de função do projeto.

2 - Resposta - O sistema lista o nome de todos os grupos de dados e processos elementares seguidos da complexidade e a quantidade de pontos de função de cada um deles de acordo com a tabela do caso de uso 21, total do somatório de todos os pontos de funções apresentados, e de acordo com o tipo de versão (Desenvolvimento, melhoria e aplicação) realiza a contagem conforme calculo abaixo:

Desenvolvimento:

$$DFP = (UFP + CFP) \times VAF$$

Onde:

DFP: é o número de pontos de função do projeto de desenvolvimento.

UFP: é o número de pontos de função não-ajustados das funções disponíveis após a instalação.

CFP: é o número de pontos de função não-ajustados das funções de conversão.

VAF: Valor do fator de ajuste.

Melhoria:

$$\text{EFP} = [(\text{ADD} + \text{CHGA} + \text{CFP}) \times \text{VAFA}] + (\text{DEL} \times \text{VAFB})$$

Onde:

EFP: é o número de pontos de função do projeto de melhoria.

ADD: é o número de pontos de função não-ajustados das funções incluídas pelo projeto de melhoria.

CHGA: é o número de pontos de função não-ajustados das funções modificadas

CFP: é o número de pontos não-ajustados adicionados pela conversão.

VAFA: é o valor do fator de ajuste da aplicação depois do projeto de melhoria.

DEL: é o número de pontos de função não-ajustados das funções excluídas pelo projeto de melhoria.

VAFB: é o valor do fator de ajuste da aplicação antes do projeto de melhoria.

Aplicação:

Se for não houver uma versão de melhoria anterior:

$$\text{AFP} = \text{ADD} \times \text{VAFs}$$

Onde:

AFP: é o número de pontos de função ajustados da aplicação.

ADD: é o número de pontos de função não-ajustados das funções instaladas.

VAF: é o valor do fator de ajuste da aplicação.

Formula após projeto de melhoria:

$$\text{APF} = [(\text{UFPB} + \text{ADD} + \text{CHGA}) - (\text{CHGB} + \text{DEL})] \times \text{VAFA}$$

Onde:

AFP: é o número de pontos de função ajustados da aplicação.

UFPB: Pontos de função não-ajustados da aplicação antes de melhoria.

ADD: Pontos de função não-ajustados das funções incluídas pelo projeto de melhoria.

CHGA: Pontos de função não-ajustados das funções alteradas pelo projeto de melhoria depois de seu término.

CHGB: Pontos de função não-ajustados das funções alteradas pelo projeto de melhoria antes do seu término.

DEL: Pontos de função não-ajustados das funções excluídas pelo projeto de melhoria.

VAF: Valor do fator de ajuste depois do projeto de melhoria.

O valor de ajuste é calculado da seguinte forma:

$$\text{VAF} = (\text{TDI} \times 0,01) + 0,65$$

Onde:

VAF: é o valor de ajuste

TDI: somatório dos pesos das respostas das características gerais.

Número: 19

Caso de uso: Consultar Mão-de-obra.

Descrição: Esta ação permite a consulta de mão-de-obra cadastrados na base de dados.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:

- **Curso Normal:**

1 - Ação - O usuário solicita consulta de mão-de-obra, podendo filtrá-lo pelo nome, e-mail ou cargo.

2 - Resposta - Se o usuário filtrar a consulta, o sistema deverá listar somente os usuários que atendam os filtros escolhidos e exibir as informações abaixo listadas:

- Nome

- E-mail

- Cargo
- Telefone
- Botão de cadastro
- Botão de alteração
- Botão de Exclusão

3 - Ação - O usuário seleciona o botão de cadastro de mão-de-obra.

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 20.

- Curso Alternativo 1:

3 - Ação - O usuário seleciona o botão de alteração de mão-de-obra

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 21.

- Curso Alternativo 2:

3 - Ação - O usuário seleciona o botão de exclusão de mão-de-obra.

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 22.

- Curso Alternativo 3:

2 - Resposta - Se o usuário não filtrar a consulta, o sistema deverá exibir toda mão-de-obra cadastrados.

- Curso Alternativo 4:

2 - Resposta - Se não houver mão-de-obra cadastrada, exibir a mensagem ao usuário: "Nenhuma mão-de-obra cadastrado" e disponibilizar o cadastro de mão-de-obra seguindo o curso normal dos eventos descrito no caso de uso

Número: 20

Caso de uso: Cadastrar mão-de-obra.

Descrição: Esta ação permite o cadastro de mão-de-obra.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário solicita cadastro de mão-de-obra.

2 - Resposta - O sistema constrói formulário com os seguintes dados:

- Nome

- e-mail

- telefone

- cargo

3 - Ação - O usuário informa os dados requeridos.

4 - Resposta - O sistema valida os dados informados.

5 - Resposta - Se os dados forem válidos, o cadastro será efetivado e a seguinte mensagem será exibida: "Mão-de-obra cadastrado com sucesso".

- Curso Alternativo:

5 - Resposta - Se os dados não forem válidos, emitir a mensagem: "Campo X inválido".

Número: 21

Caso de uso: Alterar mão-de-obra.

Descrição: Permite atualização de mão-de-obra.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:**- Curso Normal:**

1 - Ação - O usuário solicita atualização de mão-de-obra.

2 - Resposta - O sistema exibe o formulário com os dados descritos no curso normal de eventos do caso de uso 20, preenchidos.

3 - Ação - O usuário informa os dados requeridos.

4 - Resposta - O sistema valida os dados informados.

5 - Resposta - Se os dados forem válidos, o cadastro será efetivado e a seguinte mensagem será exibida: "Mão-de-obra cadastrado com sucesso".

- Curso Alternativo:

5 - Resposta - Se os dados não forem válidos, emitir a mensagem: “Campo X inválido”.

Número: 22

Caso de uso: Excluir mão-de-obra.

Descrição: Permite a exclusão de mão-de-obra.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário seleciona a exclusão de mão-de-obra.

2 - Resposta - O sistema verifica se há ligações de outros registros ao registro que será excluído. Se não houver, registra a (s) exclusão (ões), e exibe a mensagem: “Dados excluídos com sucesso”.

- Curso Alternativo:

2 - Resposta - Se identificar que existem ligações com o registro que se deseja excluir, o sistema não efetiva a exclusão e deverá emitir a mensagem: “Registro não pôde ser removido, pois existem outros registros que fazem referência a ele”.

Número: 23

Caso de uso: Consultar Categoria.

Descrição: Esta ação permite a consulta de Categoria cadastradas na base de dados.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário solicita consulta de Categoria, podendo filtrá-lo pelo nome e descrição.

2 - Resposta - Se o usuário filtrar a consulta, o sistema deverá listar somente as Categoria que atendam os filtros escolhidos e exibir as informações abaixo listadas:

- Nome
- Descrição
- Botão de cadastro
- Botão de alteração
- Botão de Exclusão

3 - Ação - O usuário seleciona o botão de cadastro de Categoria.

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 24.

- Curso Alternativo 1:

3 - Ação - O usuário seleciona o botão de alteração de Categoria

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 25.

- Curso Alternativo 2:

3 - Ação - O usuário seleciona o botão de exclusão de Categoria.

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 26.

- Curso Alternativo 3:

2 - Resposta - Se o usuário não filtrar a consulta, o sistema deverá exibir toda Categoria cadastrada.

- Curso Alternativo 4:

2 - Resposta - Se não houver Categorias cadastrada, exibir a mensagem ao usuário: "Nenhuma Categoria cadastrado" e disponibilizar o cadastro de Categoria seguindo o curso normal dos eventos descrito no caso de uso

Número: 24

Caso de uso: Cadastrar Categoria.

Descrição: Esta ação permite o cadastro de Categoria.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário solicita cadastro de Categoria.

2 - Resposta - O sistema constrói formulário com os seguintes dados:

- Nome

- Descrição

3 - Ação - O usuário informa os dados requeridos.

4 - Resposta - O sistema valida os dados informados.

5 - Resposta - Se os dados forem válidos, o cadastro será efetivado e a seguinte mensagem será exibida: "Categoria cadastrada com sucesso".

- Curso Alternativo:

5 - Resposta - Se os dados não forem válidos, emitir a mensagem: "Campo X inválido".

Número: 25

Caso de uso: Alterar Categoria.

Descrição: Permite atualização da Categoria.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário solicita atualização de Categoria.

2 - Resposta - O sistema exibe o formulário com os dados descritos no curso normal de eventos do caso de uso 24, preenchidos.

3 - Ação - O usuário informa os dados requeridos.

4 - Resposta - O sistema valida os dados informados.

5 - Resposta - Se os dados forem válidos, o cadastro será efetivado e a seguinte mensagem será exibida: "Categoria cadastrado com sucesso".

- Curso Alternativo:

5 - Resposta - Se os dados não forem válidos, emitir a mensagem: “Campo X inválido”.

Número: 26

Caso de uso: Excluir Categoria.

Descrição: Permite a exclusão de Categoria.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:**- Curso Normal:**

1 - Ação - O usuário seleciona a exclusão de Categoria.

2 - Resposta - O sistema verifica se há ligações de outros registros ao registro que será excluído. Se não houver, registra a (s) exclusão (ões), e exibe a mensagem: “Dados excluídos com sucesso”.

- Curso Alternativo:

2 - Resposta - Se identificar que existem ligações com o registro que se deseja excluir, o sistema não efetiva a exclusão e deverá emitir a mensagem: “Registro não pôde ser removido, pois existem outros registros que fazem referência a ele”.

Número: 27

Caso de uso: Consultar Modelo.

Descrição: Esta ação permite a consulta de Modelo cadastradas na base de dados.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:**- Curso Normal:**

1 - Ação - O usuário solicita consulta de Modelo, podendo filtrá-lo pelo nome e descrição.

2 - Resposta - Se o usuário filtrar a consulta, o sistema deverá listar somente os Modelos que atendam os filtros escolhidos e exibir as informações abaixo listadas:

- Nome
- Descrição
- Botão de cadastro
- Botão de alteração
- Botão de Exclusão

3 - Ação - O usuário seleciona o botão de cadastro de Modelo.

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 28.

- Curso Alternativo 1:

3 - Ação - O usuário seleciona o botão de alteração de Modelo

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 29.

- Curso Alternativo 2:

3 - Ação - O usuário seleciona o botão de exclusão de Modelo.

4 - Resposta - O sistema seguirá o curso normal dos eventos descrito no caso de uso 30.

- Curso Alternativo 3:

2 - Resposta - Se o usuário não filtrar a consulta, o sistema deverá exibir todos Modelos cadastrados.

- Curso Alternativo 4:

2 - Resposta - Se não houver Categorias cadastrada, exibir a mensagem ao usuário: "Nenhum Modelo cadastrado" e disponibilizar o cadastro de Modelo seguindo o curso normal dos eventos descrito no caso de uso

Número: 28

Caso de uso: Cadastrar Modelo.

Descrição: Esta ação permite o cadastro de Modelo.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário solicita cadastro de Modelo.

2 - Resposta - O sistema constrói formulário com os seguintes dados:

- Nome

- Descrição

- lista com as características cadastrada gerais no banco.

3 - Ação - O usuário seleciona uma característica.

4 - Resposta - O sistema preenche a informação da pergunta daquela característica, carregando as opções de resposta existente no banco de acordo com a tabela abaixo.

5 - Ação - O usuário seleciona as respostas

6 - Resposta - O sistema armazena a opção selecionada.

7 - Ação - O usuário informa os dados requeridos.

8 - Resposta - O sistema valida os dados informados.

9 - Resposta - Se os dados forem válidos, o cadastro será efetivado e a seguinte mensagem será exibida: "Modelo cadastrado com sucesso".

- Curso Alternativo:

9 - Resposta - Se os dados não forem válidos, emitir a mensagem: "Campo X inválido".

Número: 29

Caso de uso: Alterar Modelo.

Descrição: Permite atualização de Modelos.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:

- Curso Normal:

- 1 - Ação - O usuário solicita atualização de Modelo.
- 2 - Resposta - O sistema exibe o formulário com os dados descritos no curso normal de eventos do caso de uso 28, preenchidos.
- 3 - Ação - O usuário informa os dados requeridos.
- 4 - Resposta - O sistema valida os dados informados.
- 5 - Resposta - Se os dados forem válidos, o cadastro será efetivado e a seguinte mensagem será exibida: “Modelo cadastrado com sucesso”.

- Curso Alternativo:

- 5 - Resposta - Se os dados não forem válidos, emitir a mensagem: “Campo X inválido”.

Número: 30

Caso de uso: Excluir Modelo.

Descrição: Permite a exclusão de Modelo.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:

- Curso Normal:

- 1 - Ação - O usuário seleciona a exclusão de Modelo.
- 2 - Resposta - O sistema verifica se há ligações de outros registros ao registro que será excluído. Se não houver, registra a (s) exclusão (ões), e exibe a mensagem: “Dados excluídos com sucesso”.

- Curso Alternativo:

- 2 - Resposta - Se identificar que existem ligações com o registro que se deseja excluir, o sistema não efetiva a exclusão e deverá emitir a mensagem: “Registro não pôde ser removido, pois existem outros registros que fazem referência a ele”.

Número: 31

Caso de uso: Selecionar Projeto.

Descrição: Esta ação permite a seleção de um projeto e uma versão do mesmo para edição ou consulta de suas características.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade.

Ações/Respostas:

- **Curso Normal:**

1 - Ação - O usuário solicita consulta seleção do projeto.

2 - Resposta - O sistema deverá listar todos os projetos e as suas versões com as informações abaixo listadas:

- Nome
- Versão
- Tipo
- Categoria

3 - Ação - O usuário seleciona o projeto.

4 - Resposta - O sistema marca o projeto como o atual.

Número: 32

Caso de uso: Relacionar Mão-de-obra.

Descrição: Esta ação permite relacionar a mão-de-obra a projetos.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade e existir um projeto selecionado.

Ações/Respostas:

- **Curso Normal:**

1 - Ação - O usuário solicita consulta de mão-de-obra, podendo filtrá-lo pelo nome, e-mail ou cargo.

2 - Resposta - Se o usuário filtrar a consulta, o sistema deverá listar somente os usuários que atendam os filtros escolhidos e exibir as informações abaixo listadas:

- Nome

- E-mail
- Cargo
- Telefone
- caixa de checagem

3 - Ação - O usuário seleciona a caixa de checagem.

4 - Resposta - O sistema associa a mão-de-obra selecionada ao projeto e exibe a mensagem: “Mão de obra associada”.

Número: 33

Caso de uso: Importar Grupo de Dados.

Descrição: Esta ação permite a importação de Grupo de Dados de outros projetos.

Atores: Usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade e existir uma versão em edição de um projeto.

Ações/Respostas:

- **Curso Normal:**

1 - Ação - O usuário solicita importar de Grupo de Dados, podendo filtrá-lo pelo projeto, versão, nome, tipo ou complexidade.

2 - Resposta - Se o usuário filtrar a consulta, o sistema deverá listar somente os Grupo de Dados que atendam os filtros escolhidos e exibir as informações abaixo listadas:

- Projeto
- Versão
- Nome
- Tipo
- Complexidade

3 - Ação - O usuário seleciona o grupos de dados que deseja importar.

4 - Resposta - O sistema copia os dados do grupo de dado selecionado para o projeto atual e exibe a mensagem: “Grupo de dados importado com sucesso”.

Número: 34

Caso de uso: Usar Modelo.

Descrição: Esta ação permite utilizar modelo para sugestão de resposta para as características gerais.

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade e existir um projeto selecionado.

Ações/Respostas:

- Curso Normal:

1 - Ação - O usuário solicita utilização de modelos, podendo filtrá-lo pelo nome ou descrição.

2 - Resposta - Se o usuário filtrar a consulta, o sistema deverá listar somente os modelos que atendam os filtros escolhidos e exibir as informações abaixo listadas:

- Nome

- descrição

3 - Ação - O usuário seleciona o modelo.

4 - Resposta - O sistema associa o modelo selecionado ao projeto e exibe a mensagem: "Modelo adicionado".

Número: 35

Caso de uso: Calcular as Estimativas do Projeto

Descrição: Ação que permite ao usuário a estimativa de custo e duração do projeto

Atores: Administrador ou usuário.

Pré-condição: Estar logado no sistema e possuir permissão a essa funcionalidade e existir uma versão em edição de um projeto.

Ações/Respostas:

1 - Ação - O usuário solicita Calculo das Estimativas de custo e duração do Projeto.

2 - Resposta - O sistema lista as estimativas de custo e duração do projeto a

partir da fórmula:

DURAÇÃO DO PROJETO (H) = TAMANHO DO PROJETO EM PF * 4

CUSTO (R\$) = DURAÇÃO DO PROJETO * CUSTO MÉDIO DA EQUIPE

Levando-se em consideração que para cada ponto de função é necessário 4 horas para o desenvolvimento.

REFERÊNCIAS BIBLIOGRÁFICAS

CANDÉAS, Alcione Jandir; LOPES, Cleuton Clay Neres. **Estimativa: Uma Ferramenta para Agilizar o Dimensionamento de Projetos no SERPRO com base na metodologia de Análise de Pontos por Função.** Disponível em: <http://www.inf.ufsc.br/~sbes99/anais/Sessao-Ferramenta-Completo/03-estimativa.pdf>. Acesso em: 18 de fevereiro de 2008.

CORDEIRO, Marco Aurélio. **Métricas de Software.** Disponível em <http://www.pr.gov.br/batebyte/edicoes/2000/bb101/metricas.htm>. Acesso em 20/02/2008.

Carlos, Alan; Lemos, Douglas. **DAWAP – WEB ANALYSIS PROJECTS – FERRAMENTA DE AUXÍLIO À ANÁLISE DE PROJETOS DE SOFTWARES UTILIZANDO PONTOS DE FUNÇÃO.** Disponível em: http://www.frb.br/ciente/2006_2/BSI/BSI.CARLOS.etal.F1%20Rev.%20%2022.11.06.pdf. Acesso em 02/03/2008.

CRUZ, Marcio Schinkoeth; SIQUEIRA, Sigmar M. de; CHANG, Willian Chu. **Métricas de Pontos de Função.** Disponível em http://www.dcc.ufrj.br/~schneide/es/2000/1/trab/al21_28_29/PFPART1.HTM. Acesso em 05/03/2008.

DEKKERS, Carol A. **Pontos de Função e Medidas – O que é um Ponto de Função?.** Disponível em <http://www.bfpug.com.br/artigos/dekkers-pontosdefuncaoemedidas.htm> Acessado em 20/02/2008.

DIAS, Raquel. **Análise por Pontos de Função: Uma Técnica para Dimensionamento de Sistemas de Informação.** Disponível em www.facecla.com.br/revistas/resi/edicoes/ed3artigo02.pdf. Acesso em 25/02/2008

FILHO, Wilson de Pádua Paula. **Engenharia de Software – Fundamentos, Métodos e Padrões.** 2º Ed. Editora LTC.

HAZAN, Claudia. **Uma ferramenta na busca da excelência Medições de Software.** Disponível em <http://www.serpro.gov.br/imprensa/publicacoes/tematec/tematec/2004/ttec75>. Acesso em 20/02/2008.

HAZAN, Claudia; LEITE, Julio Cesar Sampaio do Prado. **Indicadores para a Gerência de Requisitos.** Disponível em http://wer.inf.puc-rio.br/WERpapers/artigos/artigos_WER03/claudia_hazan.pdf. Acesso em 05/03/2008.

IFPUG. **Manual de Práticas de Contagem de Pontos de Função , Versão 4.2.1. 2005 - International Function Point Users Group.** Disponível em

<http://www.ifpug.org>. Acesso em 21/02/2008.

MELO, Marco Aurélio Vilaça de. **Requisitos de Ferramentas de Apoio aos Processos de Medição de Software**. Disponível em <http://homepages.dcc.ufmg.br/~rodolfo/dcc823-2-07/Entrega2/Marco2.pdf> . Acesso em 21/02/2008.

MORRIS, Pam. **O Padrão ISO para Medição Funcional de Tamanho**. Disponível em: http://www.bfpug.com.br/Artigos/iso_fsm.htm. Acesso em 19/02/2008.

PETERS, James F.; PEDRYCZ, Wiltold. **Engenharia de Software – Teoria e Prática**. 3ª Tiragem. Editora Campus.

PRESSMAN, Roger S., **Engenharia de Software**. São Paulo, Pearson Education do Brasil, 1995.

TAVARES, Helena Cristina A. B.; CARVALHO, Ana Elizabete S.; CASTRO, Jaelson F. B. **Medição de Pontos por Função a Partir da Especificação de Requisitos**. Disponível em: http://wer.inf.pucrio.br/WERpapers/artigos/artigos_WER02/tavares.pdf. Acesso em: 19 de fevereiro de 2008.

VAZQUEZ, Carlos Eduardo; SIMÕES, Guilherme Siqueira; ALBERT, Renato Machado. **Análise de Pontos de Função: Medição, Estimativas e Gerenciamento de Projetos de Software**. 2ª ed. Editora Érica Ltda