



**Pontifícia Universidade Católica de Minas Gerais  
INSTITUTO DE EDUCAÇÃO CONTINUADA  
GERÊNCIA DE PROJETO DE SOFTWARE**

**AVALIAÇÃO DE UM PROCESSO EMPÍRICO PARA ESTIMAR O  
ESFORÇO DE PROJETOS DE SOFTWARE**

Daniel Albricker de Carvalho

Belo Horizonte  
2009

**Daniel Albricker de Carvalho**

**AVALIAÇÃO DE UM PROCESSO EMPÍRICO PARA ESTIMAR O  
ESFORÇO DE PROJETOS DE SOFTWARE**

Monografia de conclusão do curso de pós-graduação *lato sensu* Gerência de Projetos de Software, necessária para obtenção do grau de especialista.

Orientador: Rogério Baldini

Belo Horizonte

2009

## **AGRADECIMENTOS**

Gostaria de agradecer a Deus e minha querida esposa Karine, por serem meus principais motivadores para conclusão deste trabalho.

Agradeço também aos meus pais e minhas irmãs, pelo apoio incondicional em todos os momentos de minha vida.

Agradeço ao meu orientador Rogério Baldini pelos ensinamentos e principalmente pela paciência durante a realização deste trabalho.

Enfim agradeço ao colega e amigo Alexandre Oliveira, gerente da Fábrica de Software da TOTVS – BH, pelos incentivos, sugestões e críticas, que foram fundamentais para que eu concluísse a especialização.

## RESUMO

Esta monografia avalia o processo empírico de estimativa do esforço de projetos de software implantado pela Fábrica de Software da Empresa TOTVS, unidade Belo Horizonte, com o objetivo de fornecer uma perspectiva em relação à eficiência e confiabilidade deste processo utilizado pela empresa. Este trabalho apresenta um levantamento bibliográfico sobre o processo de estimativas abordado no CMMI – Capability Maturity Model Integration e sobre a aplicação da Análise de Pontos de Função, ferramenta de estimativa de tamanho de software cada vez mais utilizada como apoio em diversas áreas da gerência de projetos. O estudo de caso desta monografia aborda a metodologia de desenvolvimento de software implantada na Fábrica de Software da TOTVS – BH, no qual é detalhado e exemplificado o processo de estimativa que ocorre durante a fase de levantamento de requisitos dos projetos da empresa, e a avaliação deste processo, comparando-o com um processo formal de estimativa de software, a partir da análise de pontos de função.

Palavras-chave: Gerenciamento de Projeto; Processo de Software; Metodologia; Métricas, Pontos de Função.

## **ABSTRACT**

This paper evaluates the empiric process of effort estimation for software projects by Software Factory at TOTVS' company, in Belo Horizonte, with the objective to provide a perspective of efficiency and thrust of this process. This paper presents a theoretical study about estimation process approached in CMMI - Capability Maturity Model Integration and about the application of Function Points Analysis. FPA is the effort estimation technique more used in many areas of project management. The case studied in this paper covers software development methodology implanted on Software Factory of TOTVS, whose estimation process occurs during the requirement analysis stage. This empirical process is compared with a formal software estimation process, the Function Point Analysis.

Key-words: Project Management, Software Process; Methodology; Metrics; Function Points.

## LISTA DE FIGURAS

Figura 1 – Processos da contagem de pontos de função .....	21
Figura 2 – Casos de Uso modelados pelo Analista de Customização a partir dos requisitos.....	40
Figura 3 – Diagrama de Atividades do Caso de Uso “Cadastrar os Embalagens e os respectivos preços por produto” .....	40
Figura 4 – Botão Customizado da Visão de Produtos.....	41
Figura 5 – Cadastro Customizado de embalagens por produto .....	42
Figura 6 – Diagrama de Atividade do Caso de Uso “Inserir Tipos de Embalagens X Quantidade por item de movimento” .....	43
Figura 7 – Botão Customizado no cadastro de Itens de Movimento .....	45
Figura 8 – Tela de Quantidade de Vasilhames por produto .....	45

## LISTA DE TABELAS

Tabela 1 – Complexidade funcional dos ALI e AIE .....	24
Tabela 2 – Contribuição dos pontos de função não ajustados das funções do tipo dado .....	24
Tabela 3 – Tabela de Complexidade para Entradas Externas (EE) .....	27
Tabela 4 – Tabela de Complexidade para Saídas Externas (SE) e Consultas Externas (CE).....	27
Tabela 5 – Contribuição dos pontos de função não ajustados das funções do tipo transação.....	27
Tabela 6 – Tabela de níveis de influência das características gerais do sistema .....	28
Tabela 7 – Tipos dos Requisitos ou Atividades de customização .....	36
Tabela 8 – Graus de dificuldade dos requisitos.....	37
Tabela 9 – Matriz de Esforço.....	37
Tabela 10 – Estrutura da tabela ZPRECOPRODUTO .....	42
Tabela 11 – Planilha de atividades do projeto de customização .....	46
Tabela 12 – Tabela de apropriação do esforço para cada atividade.....	47
Tabela 13 – Tabela com os prazos modificados empiricamente pelo analista.....	48
Tabela 14 – Contagem das Funções de Dados .....	50
Tabela 15 – Contagem das Funções do tipo transação .....	51
Tabela 16 – Contagem dos níveis de influência.....	52
Tabela 17 – Apropriação dos percentuais de esforço de acordo com a atividade do ciclo de vida.....	53
Tabela 18 – Planilha de Horas do Projeto baseado em Pontos de Função .....	54

## SUMÁRIO

1. INTRODUÇÃO .....	9
1.1. Motivação .....	10
1.2. Objetivo.....	10
1.3. Definição do Problema.....	11
1.4. Contribuições .....	12
1.5. Organização da Monografia.....	12
2. REVISÃO BIBLIOGRÁFICA .....	14
2.1. Gerenciamento de Projetos .....	14
2.2. O modelo CMMI.....	15
2.3. Estimativa de Tamanho de Software através da Análise de Pontos de Função .....	18
3. FUNDAMENTAÇÃO TEÓRICA.....	19
3.1. Introdução.....	19
3.2. Análise de Pontos de Função .....	19
3.3. Cálculo do Esforço / Produtividade.....	31
4. EXPERIMENTOS E RESULTADOS .....	33
4.1. Estudo de Caso .....	33
4.1.3.2.1. Processo “Cadastrar as Embalagens e os respectivos preços por produto”.....	40
4.1.3.2.2. Processo “Inserir Tipos de Embalagens X Quantidade por item de movimento” .....	43
5. CONSIDERAÇÕES FINAIS .....	55
5.1. Análise dos Resultados .....	55
6. Referências .....	57

## 1. INTRODUÇÃO

Devido ao crescimento do mercado de Tecnologia da Informação, a competição entre as empresas de desenvolvimento de software tem aumentado a cada ano.

De acordo com Gartner Group, o mercado mundial de licença de software de gestão empresarial integrada, incluindo ERP, CRM, BI e SCM, superou a marca de US\$13,4 bilhões em 2004, esperando-se um crescimento anual de 6,4% até 2009, atingindo US\$18,3 bilhões. O crescimento contínuo no mercado mundial de software de gestão empresarial integrada deve-se, em grande parte, à globalização e à conseqüente exigência para que os negócios busquem otimizar suas operações de maneira a atender às demandas cada vez maiores e mais complexas por parte de seus clientes, revendedores e outros parceiros comerciais. (Prospecto Preliminar de Distribuição Pública Primária e Secundária de Ações Ordinárias de Emissão da TOTVS S.A., 2006).

Há uma necessidade das empresas que trabalham com projetos de software em implantar uma metodologia que potencialize a execução dos projetos, garantindo qualidade do produto final, sem que haja prejuízo com relação ao custo e cronograma previsto. Uma alternativa para melhorar esses resultados pode ser obtida através da elaboração de métricas para estimativa do tamanho do projeto no início do ciclo do desenvolvimento, pois a partir da estimativa do tamanho do software será possível calcular esforço e custo do projeto, bem como gerenciar o processo de desenvolvimento do software de forma eficiente.

Uma das técnicas mais utilizadas pelas empresas de software para medir o tamanho de um software, no que diz respeito às funcionalidades fornecidas por ele é a Análise de Pontos de Função, que começou a ser significativamente utilizada no Brasil no início da década de 90, principalmente por ter sido considerada uma medida padrão para grandes contratos públicos de desenvolvimento e manutenção de sistemas (VAZQUEZ, SIMÕES, ALBERT, 2008).

Neste trabalho será efetuado um estudo sobre o processo formal de estimativa de tamanho de software através da análise de pontos de função, e a aplicação da mesma em um projeto da Fábrica de Software do Grupo TOTVS, unidade Belo Horizonte, com o objetivo de avaliar a eficiência de um processo empírico atualmente adotado pela empresa em comparação com o processo de análise de pontos de função.

## 1.1. Motivação

O que motivou a escolha deste tema para este trabalho foi a necessidade de avaliar a metodologia para estimar o esforço dos projetos de customização na Fábrica de Software da empresa de tecnologia TOTVS, devido à crescente demanda dos projetos, o que exige o aumento na confiabilidade das estimativas realizadas.

A Fábrica de Software da TOTVS é o departamento responsável por implementar as customizações dos aplicativos desenvolvidos pela empresa, ou seja, implementar as regras de negócio específicas dos clientes, não contempladas nas versões originais dos aplicativos, vendidas normalmente em forma de produto.

Somente no 1º semestre de 2008, foram iniciados 251 projetos de customização, o que corresponde a um aumento de 37% do número de projetos iniciados no mesmo período de 2007. Devido ao aumento de vendas das soluções de gestão empresarial e o conseqüente aumento na demanda das customizações, torna-se essencial a melhoria contínua dos processos de desenvolvimento de software, dentre eles o processo de estimativas de tamanho. Além do aumento do número de projetos, notou-se um aumento expressivo na quantidade de projetos com prazos acima de 160 Horas, sendo que no 1º semestre de 2008 o índice aumentou 57%, o que implica a necessidade de aumentar ainda mais a confiabilidade das estimativas, por se tratar de projetos de média e longa duração.

## 1.2. Objetivo

O objetivo geral dessa pesquisa monográfica é:

Fazer uma avaliação de um processo empírico de estimativa de esforço do software, atualmente adotado pela Fábrica de Software da filial Belo Horizonte do Grupo TOTVS, e comparar a eficiência do mesmo, utilizando um processo formal de ponto de função, a partir de um estudo de caso.

Os objetivos específicos são:

- Estudar o processo “Estabelecer Estimativas” do gerenciamento de projetos, de acordo com o modelo SEI CMMI;

- Estudar as principais características da técnica de análise por pontos de função;
- Descrever um processo empírico de estimativa de tamanho e esforço de software, apresentando suas vantagens e desvantagens;
- Comparar o processo empírico com processo formal de análise por pontos de função.

Ao término deste trabalho será possível obter uma conclusão sobre a eficiência do processo empírico utilizado pela Fábrica de Software da TOTVS – BH e se há necessidade de alteração do modelo aplicado atualmente.

### **1.3. Definição do Problema**

Atualmente a Fábrica de Software da TOTVS – BH dispõe de uma metodologia para estimar o esforço de software nos projetos de customização, implantada em Janeiro de 2004, quando a empresa ainda era RM Sistemas<sup>1</sup>.

Após aquisição da RM Sistemas pela TOTVS, houve diversas mudanças culturais e estratégicas da empresa, como o aumento do investimento na área de serviços da empresa e um crescimento significativo da demanda de projetos de customização, o que acarretou a necessidade de reavaliar vários processos do departamento.

Diante desse contexto surgiu um problema em relação à confiabilidade do processo de estimativa de esforço dos projetos de customização, que por ser feito de forma empírica, pode não ser o mais adequado para que o departamento consiga dar vazão à elevada demanda, de forma eficiente, dentro do prazo, custo e qualidade esperados pelos clientes e pela própria empresa.

---

<sup>1</sup> A empresa RM Sistemas foi incorporada ao Grupo TOTVS S.A. em abril de 2006.

#### **1.4. Contribuições**

Espera-se que este trabalho contribua inicialmente com as equipes de desenvolvimento da fábrica de software da TOTVS, no sentido de mostrar as vantagens e desvantagens da utilização de um processo empírico para realizar estimativas, e apresentar o seu nível de eficiência, quando comparado a um processo formal de análise por pontos de função. Procurar-se-á estabelecer cenários próximos da realidade dos projetos de customização, de forma que os resultados obtidos neste trabalho possam ser avaliados pelos líderes do departamento.

Este trabalho também estende a outras empresas, podendo subsidiar decisões nas organizações que enfrentam dificuldades em estimar o tamanho de seus projetos nas fases iniciais, e que não possuem metodologia ou não sabem qual seria a melhor técnica a ser aplicada a cada projeto.

#### **1.5. Organização da Monografia**

Este trabalho será dividido nas seguintes partes:

1. Revisão Bibliográfica - Neste tópico é apresentado um levantamento bibliográfico do processo de estimativas proposto pelo CMMI, e a utilização pelo mercado da técnica de métrica de software por pontos de função;
2. Fundamentação Teórica – Este tópico descreve a técnica de análise por pontos de função, que será utilizada na aplicação prática desta monografia;
3. Experimentos e resultados – Neste tópico é apresentado o processo empírico de métricas adotado pela Fábrica de Software da TOTVS. Posteriormente apresenta-se uma comparação do processo empírico com a técnica de pontos de função em um projeto real da fábrica de software, mostrando os resultados obtidos;
4. Considerações Finais e Conclusão – O fechamento desta monografia dá-se pela conclusão extraída dos experimentos feitos e dos resultados obtidos. Espera-se ao final deste trabalho obter uma conclusão sobre a

comparação entre um processo empírico e um processo formal de estimativa de software.

## 2. REVISÃO BIBLIOGRÁFICA

### 2.1. Gerenciamento de Projetos

Segundo o Project Management Institute (PMI), um projeto é um esforço temporário utilizado para criar um produto, serviço ou resultado exclusivo. O gerenciamento de projetos é a aplicação de conhecimento, habilidades, ferramentas e técnicas às atividades do projeto com o objetivo de atender aos seus requisitos. É realizado através da aplicação de forma integrada dos seguintes processos de gerenciamento de projetos: iniciação, planejamento, execução, monitoramento e controle, e encerramento (PMI, 2004).

Segundo Hazan (2006), a partir do planejamento do projeto de software é possível ter uma visão geral dos requisitos que definem o escopo do produto, com o objetivo de estabelecer e manter planos que definam as atividades a serem realizadas durante o projeto. Uma das principais atividades durante a fase de planejamento do projeto é elaborar estimativas, pois fornecerão métricas que deverão atender às necessidades de comunicação e informação do projeto. Baseado nessas métricas, o gerente do projeto será capaz de acompanhar o projeto ao longo do seu ciclo de vida, e identificar eventuais problemas de forma ágil, direcionando esforços para solucioná-los. A análise baseada nas métricas permite que as principais variáveis do projeto sejam monitoradas, tais como cronograma, custos, qualidade, riscos e escopo (VAZQUEZ, SIMÕES, ALBERT, 2008).

As estimativas de tamanho são o primeiro tipo de análise quantitativa efetuada no projeto de software e é o processo necessário para calcular o número de períodos do trabalho que serão necessários para finalizar as atividades do projeto (PMI, 2004).

Segundo Boehm (2000, *apud* HAZAN, 2006), um dos principais riscos de um projeto é a falta de credibilidade nas estimativas pelas equipes de desenvolvimento. Isto ocorre quando as estimativas de tamanho são imprecisas, ou seja, quando os projetos são subestimados ou superestimados.

Em muitos casos, as estimativas inadequadas de tamanho de software podem levar ao estabelecimento de objetivos difíceis de se alcançar, acarretando em prazos não cumpridos, custos excessivos com horas extras e má qualidade do

software, constituindo um dos principais fatores causadores do cancelamento de projetos (HAZAN, 2006). Para tornar a atividade de desenvolvimento de projetos de software mais previsíveis, existem modelos para medir e melhorar os processos da organização, como por exemplo o CMMI - Capability Maturity Model Integration.

## **2.2. O modelo CMMI**

Segundo Vazquez, Simões e Albert (2008), o modelo CMMI é um dos modelos de maturidade mais difundidos no mundo, tendo sido originado a partir do modelo CMM, desenvolvido durante a década de 80 pelo SEI (*Software Engineering Institute*). Este modelo é dividido em 5 níveis de maturidade (Inicial, Gerenciado, Definido, Gerenciado Quantitativamente e Otimizado), onde cada nível possui um conjunto pré-definido de áreas de processos (SEI, 2002).

### **2.2.1. O processo CMMI para o estabelecimento de Estimativas**

O processo “Estabelecer Estimativas” compõe uma das principais atividades da área de planejamento do projeto de software do nível 2 no modelo CMMI (SEI,2002).

É importante ressaltar que o CMMI aborda as principais estimativas que devem ser consideradas no processo de planejamento de projetos, porém ele não define e não descreve como implantar um processo de estimativa.

De acordo com o CMMI, este processo abrange, entre outros, os seguintes sub-processos:

#### **2.2.1.1. Estabelecer estimativas de atributos de produtos de trabalho e tarefas**

Este processo aborda a estimativa do tamanho, principal entrada de muitos modelos utilizados para estimar o esforço, custo e cronograma do projeto, embora entradas como conectividade, complexidade e estrutura também podem servir de base para realizar tais estimativas (SEI, 2002). De acordo com o CMMI, para cada

atributo de tamanho deve ser atribuído um nível relativo de dificuldade ou complexidade.

As sub-práticas do CMMI que compõem o processo para estabelecer estimativas de atributos de produtos de trabalho e tarefas são as seguintes:

- a. Determinar a abordagem técnica para o projeto, ou seja, esta sub-prática aborda os requisitos não funcionais do produto e inclui as decisões tomadas em relação às características de arquitetura do sistema, tecnologia que será utilizada e abrangência esperada das funcionalidades dos produtos finais, como segurança, confiabilidade e questões ergonômicas (SEI, 2002);
- b. Usar métodos apropriados para determinar os atributos de produtos de trabalho e tarefas que serão utilizados para estimar os requisitos de recursos, ou seja, a metodologia para determinar o tamanho e a complexidade deve ser baseada em modelos válidos ou dados históricos, e deve evoluir conforme aumenta o entendimento do relacionamento das características do produto com seus atributos. Os métodos sugeridos pelo CMMI para estimar o tamanho dos requisitos incluem os métodos de contagem de Pontos de função para software e contagem de Pontos por Caso de Uso (SEI, 2002);
- c. Estimar os atributos dos produtos de trabalho e tarefas;
- d. Estimar, conforme apropriado, o trabalho, maquinário, materiais e métodos que serão exigidos pelo projeto.

Os principais artefatos que compõem este sub-processo são: abordagem técnica, tamanho e complexidade das tarefas e produtos de trabalho, modelos de estimativas e estimativas de atributos.

#### **2.2.1.2. Determinar estimativas de Esforço e Custo**

De acordo com o modelo CMMI, as estimativas de esforço e custo devem ser geradas baseando-se nos modelos e dados históricos aplicados ao tamanho e outros parâmetros de planejamento. Desta forma a estimativa de esforço e custo

pode ser obtida a partir de dados históricos do esforço e custo despendido em projetos anteriores de características semelhantes, juntamente com o tamanho dimensionado nos respectivos projetos.

Pode haver casos que os dados históricos não se aplicam, quando os esforços a serem estimados são considerados “sem precedentes”, ou seja, se um produto ou tipo de tarefa semelhante ao que será estimado nunca foi construído anteriormente (SEI, 2002). Neste caso os riscos de erros nas estimativas aumentam e exigem um esforço maior para desenvolver novos critérios de estimativas para a nova característica desse produto ou tarefa.

As sub-práticas do CMMI que compõem o processo para determinar as estimativas de esforço e custo dos projetos são as seguintes:

- a. Coletar modelos ou dados históricos que serão utilizados para transformar os atributos dos produtos de trabalho e tarefas em estimativas de horas de trabalho e custo, cujos dados históricos incluem os dados de custo, esforço e cronograma de projetos executados anteriormente, além de dados apropriados de escala para equilibrar as diferenças de tamanho e complexidade (SEI, 2002).
- b. Incluir necessidades de infra-estrutura de suporte quando estiver estimando o esforço e o custo, onde na engenharia de software, devem ser identificados os recursos computacionais críticos no ambiente servidor, ambiente de testes ou no ambiente de produção do projeto, e basear as estimativas de recursos computacionais nos requisitos alocados. Exemplos de recursos computacionais incluem: memória, processador, espaço em disco, periféricos, capacidade da rede e do banco de dados (SEI, 2002).
- c. Estimar o esforço e o custo utilizando modelos e/ou dados históricos, cujas principais entradas para realizar estimativas incluem, dentre outras: riscos, competências críticas e papéis necessários para executar o trabalho, requisitos de produtos, abordagem técnica, nível de habilitação de gerentes e pessoal para executar o trabalho, conhecimento, habilidades e treinamento, viagens, etc (SEI, 2002).

### 2.3. Estimativa de Tamanho de Software através da Análise de Pontos de Função

Atualmente existem várias técnicas para estimativa de tamanho desenvolvidas com base nas funcionalidades do software, tais como: Mark II, COSMIC-FFP, NESMA, Análise de Pontos de Caso de Uso e Análise de Pontos de Função, sendo a última a mais aceita e utilizada pelo mercado de software (VAZQUEZ, SIMÕES, ALBERT, 2008).

A Análise de Pontos de Função (APF) é a técnica que mede as funcionalidades fornecidas por um software sob o ponto de vista do usuário. Surgiu como uma alternativa às métricas baseadas em linhas de código e foi introduzida na década de 70, por Alan Albrecht, na época funcionário da IBM (VAZQUEZ, SIMÕES, ALBERT, 2008).

A contagem dos pontos de função é regulamentada pelo International Function Point Users Group (IFPUG), entidade sem fins lucrativos, sediada nos Estados Unidos e composta por pessoas e empresas de diversos países, cujo método foi oficializado através do padrão internacional ISO/IEC 20926 de 2002 (VAZQUEZ, SIMÕES, ALBERT, 2008).

Segundo o *Brazilian Function Point Users Group* (BFPUG), representação brasileira oficial do IFPUG, ponto de função é uma métrica funcional de tamanho de software, baseada em uma avaliação padronizada dos requisitos lógicos fornecidos ao usuário. Uma métrica funcional de tamanho é uma medida externa, pois mede somente aquilo que é percebido pelos usuários do produto de software, independentemente da forma com que os requisitos serão implementados (tecnologia utilizada, experiência da equipe, etc) (BFPUG).

No Brasil a técnica de análise de pontos de função começou a ser significativamente utilizada no início da década de 90, porém o interesse do mercado consolidou-se quando passou a ser utilizada em grandes contratos públicos de desenvolvimento de sistemas (VAZQUEZ, SIMÕES, ALBERT, 2008).

### **3. FUNDAMENTAÇÃO TEÓRICA**

#### **3.1. Introdução**

Segundo Vazquez, Simões e Albert (2008), o processo de estimativa tem o propósito inicial de fornecer informações que beneficiam o planejamento e controle dos projetos. Este processo envolve, basicamente, as seguintes atividades:

- Estimar o tamanho do produto a ser gerado;
- Estimar o esforço empregado na execução do projeto;
- Estimar a duração do projeto;
- Estimar o custo do projeto.

Para obter tais informações é necessário primeiramente escolher a unidade que será utilizada para medir o tamanho do produto. Fazendo uma analogia com projetos da construção civil, a unidade de medida de tamanho padrão para esses tipos de projeto é o metro quadrado ( $m^2$ ), que serve como unidade base para calcular custos, enquanto que na engenharia de software, entre outras unidades de medida de tamanho do software observa-se que o ponto de função tem sido a unidade mais utilizada pelo mercado (VAZQUEZ, SIMÕES, ALBERT, 2008).

Este capítulo descreve uma visão geral do processo de contagem de pontos de função, abordando a definição dos termos básicos para aplicação da técnica, pois a teoria contida neste capítulo será aplicada no estudo de caso descrito no capítulo seguinte. Este capítulo também apresenta a relação do ponto de função como medida básica para cálculo do esforço do projeto de software.

#### **3.2. Análise de Pontos de Função**

Segundo Vazquez, Simões e Albert (2008), a análise de pontos de função é uma técnica que mede o tamanho funcional do software, sob o ponto de vista do usuário, e que possui os seguintes objetivos primários:

- Medir as funcionalidades do software solicitadas e recebidas pelo usuário;
- Medir o desenvolvimento e a manutenção do software, independente da tecnologia utilizada durante a implementação.

### **3.2.1. Benefícios**

Vazquez, Simões e Albert (2008) destacam diversos benefícios da aplicação da análise de pontos de função nas empresas:

- É uma ferramenta que determina o tamanho de um software adquirido por uma empresa através da contagem de todas as funcionalidades incluídas no sistema;
- Apóia à análise de produtividade e qualidade, seja diretamente ou de forma conjunta com outras métricas de esforço, defeitos e custo;
- Apóia o gerenciamento do escopo de projetos, uma vez que podem ser feitas diversas medições por pontos de função ao longo do ciclo de vida do projeto, possibilitando determinar se os requisitos tiveram variação de tamanho e se tal variação corresponde à inclusão de novos requisitos ou alteração de requisitos já existentes;
- Auxilia o gerenciamento de requisitos, pois reforça a clareza e nível de detalhamento dos requisitos especificados;
- Serve como base para estimar custo e recursos em projetos de software, uma vez que a estimativa de pontos de função no início do ciclo de vida do projeto pode ser utilizada como entrada para outros modelos de esforço, custo e prazo.
- Pode auxiliar na negociação de contratos, pois permite o estabelecimento de preço de contratos por cada unidade de ponto de função, já que é uma medida que representa um fator tangível para o cliente.

### **3.2.2. Processo de contagem de pontos de função**

O processo para contagem de pontos de função abrange 7 etapas relacionadas entre si, como mostra o diagrama da Figura 1:

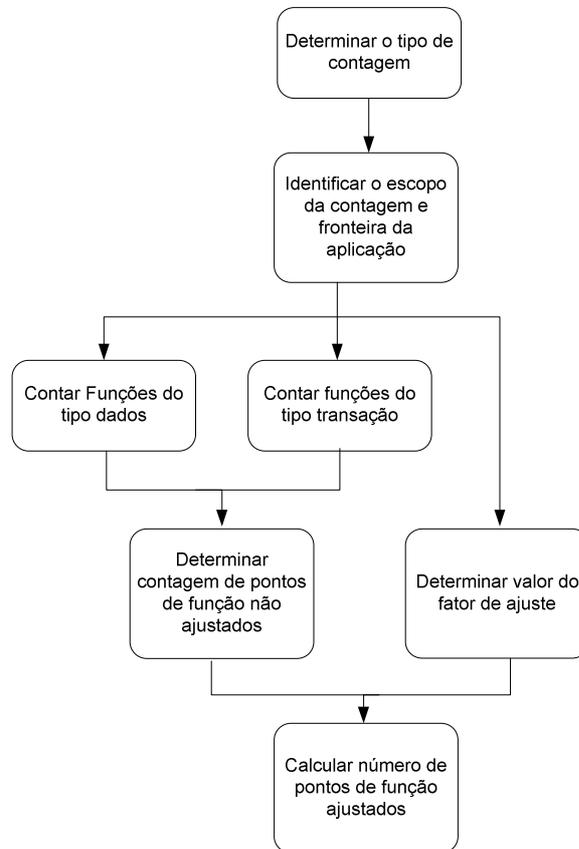


Figura 1 – Processos da contagem de pontos de função  
Fonte: Vazquez, Simões e Albert, 2008

As seções seguintes descrevem cada etapa do processo de contagem de pontos de função.

### 3.2.2.1. Determinar o tipo de contagem

Neste processo, os responsáveis pela medição determinam o tipo da contagem que será utilizada para medir o software, podendo ser um dos 3 tipos descritos a seguir (VAZQUEZ, SIMÕES, ALBERT, 2008):

- Contagem de um projeto de desenvolvimento: Este tipo de contagem mede as funcionalidades fornecidas aos usuários de acordo com os requisitos levantados no início do projeto do software. Ressalta-se que no

escopo deste tipo de contagem devem ser incluídas as funções de migração de dados do sistema que será substituído para o novo sistema, caso haja necessidade desta migração.

- Contagem de um projeto de melhoria: Este tipo de contagem mede as funcionalidades acrescentadas, modificadas ou excluídas de um projeto de melhoria para um sistema existente.
- Contagem de uma aplicação instalada: Também chamado de pontos de função instalados ou *baseline*, mede as funcionalidades de um sistema instalado, após o término do projeto que originou ou modificou o respectivo sistema.

### **3.2.2.2. Identificar o escopo da contagem e fronteira da aplicação**

Segundo Vaquez (2008), a fronteira da aplicação corresponde à definição do limite do software que será medido com o seu mundo exterior (usuários e outras aplicações), em outras palavras, delimita onde uma aplicação termina e outra começa. A identificação da fronteira da aplicação deve ser feita com base no ponto de vista do usuário e na distinção das funcionalidades conforme estabelecido nos processos do negócio, e não em considerações tecnológicas.

O escopo define as funcionalidades que serão incluídas na contagem, e pode abranger um ou mais sistemas ou apenas parte de um sistema. Pode incluir todas as funcionalidades disponíveis, todas as funcionalidades que serão utilizadas pelo usuário ou apenas algumas funcionalidades específicas, sendo esta mais comum em projetos de melhoria (VAZQUEZ, SIMÕES, ALBERT, 2008).

### **3.2.2.3. Contar funções do tipo dados**

De acordo com Vazquez, Simões e Albert (2008), as funções do tipo dado correspondem às funcionalidades fornecidas ao usuário que atendem suas necessidades de dados internos e externos à aplicação. Elas podem ser classificadas em Arquivo Lógico Interno e Arquivo de Interface Externa, e são definidas da seguinte forma:

- Arquivo Lógico Interno (ALI): grupo de dados ou informações de controle logicamente relacionado, reconhecido pelo usuário e mantido dentro da fronteira da aplicação que será medida;
- Arquivo de Interface Externa (AIE): grupo de dados ou informações de controle logicamente relacionado, reconhecido pelo usuário e mantido fora da fronteira da aplicação que será medida.

Para determinar a complexidade de cada arquivo lógico interno e arquivo de interface externa, primeiramente devem ser calculadas as quantidades dos tipos de dados e tipos de registros de cada arquivo.

Tipo de dado (TD) é um campo único e não repetido, reconhecido pelo usuário, enquanto que tipo de registro (TR) é um subconjunto de dados, reconhecido pelo usuário, pertencente a um ALI ou AIE.

#### Regras de contagem de Tipos de Dados:

- Contar um tipo de dado para cada campo único reconhecido pelo usuário e não repetido, mantido por um ALI ou AIE. Um campo reconhecido pelo usuário, por exemplo “Data de Nascimento”, é considerado apenas um TD, mesmo quando é armazenada logicamente em múltiplos campos (Dia, Mês e Ano).
- Quando existe integração entre duas ou mais aplicações, que atualizam e referenciam o mesmo ALI/AIE, deve ser contado somente os campos utilizados pela aplicação em análise.
- Contar um tipo de dado para cada campo solicitado pelo usuário que relacione com outro arquivo lógico (ALI ou AIE). Exemplo: chaves estrangeiras de entidades relacionais.

#### Regras de contagem de Tipos de Registros:

- Contar um tipo de registro para cada subgrupo de um ALI ou AIE;
- Caso não haja nenhum subgrupo, o próprio ALI ou AIE deverá ser contado como um TR.

Após obter as contagens dos tipos de dados e tipos de registros, a classificação com relação à complexidade ficará de acordo com a Tabela 1:

Número de Tipos de Registros	Número de Tipos de Dados		
	De 1 a 19	De 20 a 50	51 ou mais
Apenas 1	Baixa	Baixa	Média
De 2 a 5	Baixa	Média	Complexa
6 ou mais	Média	Complexa	Complexa

Tabela 1 – Complexidade funcional dos ALI e AIE  
Fonte: Vazquez, Simões e Albert, 2008

Após a determinação da complexidade de cada função do tipo dado, o número de pontos de função não ajustados é calculado de acordo com relação entre o Tipo da Função e sua complexidade.

Conforme descrito na Tabela 2 a seguir, um Arquivo Lógico Interno irá contribuir com 7, 10 ou 15 Pontos de Função caso seja de complexidade baixa, média ou alta, respectivamente. Para um Arquivo de Interface Externa, o número de pontos de função será 5, 7 ou 10 respectivamente para as complexidades baixa, média ou alta.

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

Tabela 2 – Contribuição dos pontos de função não ajustados das funções do tipo dado  
Fonte: Vazquez, Simões e Albert, 2008

#### **3.2.2.4. Contar funções do tipo transação**

As funções do tipo transação são as funcionalidades de processamento de dados da aplicação, fornecidas ao usuário. Elas podem ser classificadas em Entradas Externas, Saídas Externas e Consultas Externas, e são definidas da seguinte maneira, segundo Vazquez, Simões e Albert (2008):

- Entrada Externa (EE): É um processo elementar, ou seja, a menor unidade de atividade significativa para o usuário, que processa dados ou informações de controle, recebidos de fora da fronteira da aplicação, cuja

principal função é manter um ou mais ALIs ou inclusive modificar o comportamento do sistema. Exemplo: Janela que permite adicionar, excluir e alterar registros em ALI representam três entradas externas.

- Saída Externa (SE): Processo elementar que envia dados ou informações de controle para fora da fronteira da aplicação, cuja função principal é fornecer uma informação ao usuário através da lógica de processamento, seja por meio de fórmula matemática ou cálculo. Esse processo pode gerar dados derivados ou manter um ou mais ALI. Exemplos de Saídas Externas: Relatórios com totalização de dados, consultas com cálculos ou apresentação de dados derivados, arquivo de remessa gerado para outra aplicação, informações em formato gráfico, telas de login com criptografia.
- Consulta Externa (CE): Assim como a Saída Externa, também é um processo elementar que envia dados ou informações de controle para fora da fronteira da aplicação, por meio de uma simples recuperação de dados de ALIs ou AIEs. A lógica de processamento não deve possuir fórmulas matemáticas ou cálculos e nenhum ALI é mantido durante o processamento. Exemplo: Informações em formato gráfico sem que haja cálculos, *drop-downs* cujos dados sejam recuperados de ALI, telas de login sem criptografia, menus gerados dinamicamente de acordo com parametrização da aplicação.

Para determinar a complexidade funcional de cada Entrada Externa, Saída Externa e Consulta Externa, primeiramente devem ser calculadas as quantidades dos Arquivos Referenciados (AR) e Tipos de Dados (TD) e de cada função transacional.

Arquivo Referenciado (AR) é um Arquivo Lógico Interno mantido pela função do tipo transação ou um Arquivo de Interface Externa lido por uma função do tipo transação.

Vazquez, Simões e Albert (2008) definem as seguintes regras de contagem para Arquivo Referenciado:

- Contar um arquivo referenciado para cada ALI mantido;

- Contar apenas um arquivo referenciado para cada ALI que seja tanto mantido quanto lido;
- Contar um arquivo referenciado para cada ALI ou AIE lido durante o processamento;
- Arquivos não classificados como ALI ou AIE não devem ser contados.

Para contar os Tipos de Dados, que correspondem a cada campo não repetido reconhecido pelo usuário nas funções do tipo transação, Vazquez, Simões e Albert (2008) descrevem as seguintes regras:

- Contar um tipo de dado para cada campo não repetido e reconhecido pelo usuário, utilizado na execução de um processo;
- Contar apenas uma vez um campo que tanto entra quanto sai pela fronteira da aplicação;
- Não contar um campo que seja recuperado de um ALI ou AIE durante um processo elementar, mas não atravessa a fronteira da aplicação;
- Um campo que seja derivado do sistema e armazenado num ALI por meio de um processo elementar também não deve ser contado. Ex: Identificador de uma tabela, que não é mostrado ao usuário.
- Contar um único tipo de dado para o conjunto de mensagens (ex: mensagem de erro, confirmação de execução, etc) que podem ser apresentadas ao usuário após o término do processamento.
- Contar um tipo de dado para a capacidade de selecionar uma ação a ser tomada, mesmo que haja mais de uma maneira de ativar o mesmo processo. Exemplo: Deve ser contado um único tipo de dado para a opção de salvar um registro numa tela de cadastro, mesmo quando este processo pode ser feito clicando no botão “Salvar” ou utilizando uma tecla de atalho (CTRL + S).
- Não contar literais como tipos de dados, como por exemplo títulos de relatórios, cabeçalhos de colunas, textos para identificação de telas e campos.

- Não contar variáveis de paginação ou campos automáticos gerados pelo sistema, como por exemplo contador de registros selecionados pelo usuário numa grid.

Após obter as contagens dos Arquivos Referenciados e Tipos de Dados para cada função do tipo de transação encontrada na aplicação, a classificação com relação à complexidade ficará de acordo com a Tabela 3, para as Entradas Externas e de acordo com a Tabela 4, para as Saídas Externas e Consultas Externas:

Número de Arquivos Referenciados	Número de Tipos de Dados		
	De 1 a 4	De 5 a 15	16 ou mais
0 ou 1	Baixa	Baixa	Média
2	Baixa	Média	Complexa
3 ou mais	Média	Complexa	Complexa

Tabela 3 – Tabela de Complexidade para Entradas Externas (EE)  
Fonte: Vazquez, Simões e Albert, 2008

Número de Arquivos Referenciados	Número de Tipos de Dados		
	De 1 a 5	De 6 a 19	20 ou mais
0 ou 1	Baixa	Baixa	Média
2 ou 3	Baixa	Média	Complexa
4 ou mais	Média	Complexa	Complexa

Tabela 4 – Tabela de Complexidade para Saídas Externas (SE) e Consultas Externas (CE)  
Fonte: Vazquez, Simões e Albert, 2008

O valor do número de pontos de função é calculado de acordo com relação entre o Tipo de Função e sua complexidade, conforme descrito na Tabela 5:

Tipo de Função	Baixa	Média	Alta
Entrada Externa	3 PF	4 PF	6 PF
Saída Externa	4 PF	5 PF	7 PF
Consulta Externa	3 PF	4 PF	6 PF

Tabela 5 – Contribuição dos pontos de função não ajustados das funções do tipo transação  
Fonte: Vazquez, Simões e Albert, 2008

### 3.2.2.5. Determinar o valor do Fator de Ajuste

O Fator de Ajuste corresponde a uma variável calculada com base em 14 características gerais de sistema, que influenciam a contagem final dos pontos de função (VAZQUEZ, SIMÕES, ALBERT, 2008).

Segundo Vazquez, Simões e Albert (2008), quando a técnica foi criada ainda não havia as 14 características gerais do sistema para gerar o Fator de Ajuste. Ele era determinado de forma subjetiva, e poderia produzir uma variação de +-25% nos pontos de função não ajustados. As 14 características gerais foram introduzidas em 1984, e o Fator de Ajuste poderia produzir uma variação de +-35%.

Cada característica possui um nível de influência sobre a aplicação que pode variar de 0 a 5 a Tabela 6, apresentada a seguir mostra os níveis de influência que podem ser aplicados a cada característica:

<b>Nível de Influência</b>	<b>Descrição</b>
0	Nenhuma influência
1	Pouca influência
2	Influência moderada
3	Influência Média
4	Influência Significativa
5	Grande Influência

Tabela 6 – Tabela de níveis de influência das características gerais do sistema  
Fonte: Vazquez, Simões e Albert, 2008

As 14 características gerais do sistema são descritas a seguir:

- **Comunicação de Dados:** Descreve o nível de comunicação em que os dados ou informações de controle da aplicação são enviados e recebidos. Segundo Vazquez, Simões e Albert (2008), os níveis de 0 a 3 são classificados para aplicações *batch*, aumentando o nível conforme surge algum tipo de entrada ou saída de informação remota. Aplicações do tipo cliente/servidor normalmente devem ser pontuadas com nível 4 e para aplicações 3 camadas ou com mais de um tipo de protocolo de comunicação pontuam com nível 5.
- **Processamento Distribuído:** Descreve o nível em que o sistema transfere dados entre seus componentes. Segundo Vazquez, Simões e Albert (2008), um sistema de desktop isolado, cuja aplicação e o banco de dados executam localmente, deverá ser pontuado com nível 0. Um sistema N camadas pontuará com 4 e um sistema com componentes

executando em múltiplos servidores ou processadores, sendo executados dinamicamente de acordo com a disponibilidade será pontuado com nível 5.

- **Performance:** Descreve o nível cujos fatores de tempo de resposta e taxa de transações influenciam o desempenho da aplicação. Para pontuar o nível de influência dessa característica, a seguinte questão deve ser respondida, em relação ao grau de exigência do usuário: quão rápida deverá ser a aplicação e quanto isto influenciará o projeto.
- **Configuração Altamente Utilizada:** Trata-se de observações quanto ao nível de utilização de recursos computacionais que serão requeridos para a execução do sistema, ou seja, o nível desta característica mede o nível de infra-estrutura requerida para utilização do sistema.
- **Volume de Transações:** Trata o nível em que o alto volume de transações influencia o projeto e quanto o volume de transações processadas pela aplicação em um determinado intervalo de tempo afeta o projeto.
- **Entrada de Dados Online:** Trata o nível em que são efetuadas transações de entradas de dados na aplicação. Segundo Vazquez, Simões e Albert (2008), quando o sistema processar mais de 30% das transações em entradas de dados online essa característica deve ser pontuada com nível 5, o que mostra uma defasagem desta característica com a realidade atual, já que para as aplicações atuais, a maioria dos sistemas pontuará com 5.
- **Eficiência do Usuário Final:** Descreve o nível da aplicação em relação à influência de fatores humanos e usabilidade no desenvolvimento do sistema.
- **Atualização Online:** Descreve o nível em que os arquivos lógicos internos do sistema são atualizados de forma online.
- **Complexidade de Processamento:** Descreve o nível em que o processamento lógico ou matemático influencia na execução do sistema.
- **Reusabilidade:** Trata o nível em que a aplicação ou parte dela e o código fonte foram projetados, desenvolvidos e suportados para serem reaproveitados em outras aplicações.

- **Facilidade de Instalação:** Trata o nível em relação ao planejamento de implementação de ferramentas de conversão e instalação da aplicação, e se essas ferramentas foram testadas e fornecidas ao usuário.
- **Facilidade de Operação:** Esta característica descreve em que nível a aplicação atende a alguns aspectos operacionais automáticos, tais como inicialização, segurança e recuperação, com o objetivo de minimizar a necessidade de intervenções manuais, como montagem de fitas e manipulação de papel, por exemplo.
- **Múltiplos Locais:** Trata o nível em que o sistema foi especificamente projetado, desenvolvido e suportado para diferentes ambientes de hardware e software. Exemplo de ambientes de software a serem analisados: Sistemas Operacionais Windows, LINUX, etc. Exemplo de ambientes de hardware: PC/Pentium Intel, Computadores Macintosh, etc.
- **Facilidade de Mudanças:** Trata o nível em que a aplicação foi especificamente projetada e desenvolvida para facilitar futuras modificações em sua lógica de processamento e estrutura de dados.

Após serem determinados os níveis de influência das 14 características gerais, o fator de ajuste (VFA) é calculado com a seguinte fórmula:  $VFA = (TDI * 0,01) + 0,65$ , onde TDI representa o somatório dos níveis de influência das características gerais.

#### **3.2.2.6. Determinar a contagem de Pontos de Função não ajustados**

Os Pontos de Função não ajustados correspondem às funcionalidades do sistema fornecidas ao usuário, sem levar em consideração os requisitos não funcionais do software.

O valor total dos pontos de função não ajustados de uma determinada aplicação é obtido pela somatória dos pontos de função estimados para as funções do tipo dado (Arquivos Lógicos Internos e Arquivos de Interface Externa) e do tipo transação (Entradas Externas, Saídas Externas e Consultas Externas).

### **3.2.2.7. Calcular o número de Pontos de Função ajustados**

O cálculo dos Pontos de Função ajustados é a última etapa do processo de contagem de Pontos de Função.

De acordo com o IFPUG, existem três tipos de contagem de Pontos de Função ajustados, que variam de acordo com o tipo do projeto: projeto de desenvolvimento, projeto de melhoria e aplicação (VAZQUEZ, SIMÕES, ALBERT, 2008).

No estudo de caso deste trabalho a aplicação da técnica de estimativa por pontos de função será em um projeto de desenvolvimento. A fórmula utilizada para calcular os pontos de função ajustados (PFA) para este tipo de projeto é a seguinte:  $PFA = PFNA * VFA$ , onde PFNA é o número de pontos de função não ajustados e VFA é o valor do fator de ajuste, que é obtido conforme a fórmula descrita no tópico 3.2.2.5.

### **3.3. Cálculo do Esforço / Produtividade**

Um Ponto de Função é uma unidade de medida abstrata e relativa que conta o número de funcionalidades entregues ao usuário (VAZQUEZ, SIMÕES, ALBERT, 2008). Assim como a medida em metros quadrados do tamanho de uma casa não fornece a velocidade ou o tempo necessário para sua construção, o tamanho em Pontos de Função não fornece a produtividade ou o esforço necessário de desenvolvimento da aplicação que está sendo medida (DEKKERS, 1999). Por esse motivo, o cálculo do esforço do projeto pode ser feito a partir de uma média de produtividade da equipe de desenvolvimento, onde a produtividade é o número de horas gastas para implementar um ponto de função.

Segundo Aguiar (2003), existe uma iniciativa por parte de algumas empresas de tentar utilizar indicadores de mercado para produtividade, de acordo com a linguagem utilizada. Porém antes de utilizar qualquer indicador obtido do mercado, uma questão que deve ser observada para que se tenha certeza de que o indicador está adequado às características do projeto é verificar a compatibilidade dos critérios utilizados na elaboração desses indicadores de produtividade com os critérios e metodologia adotados pela empresa (VAZQUEZ, SIMÕES, ALBERT, 2008).

Uma das maiores organizações que mantém um banco de dados onde é possível obter números relacionados à produtividade com pontos de função para projetos de diversos contextos tecnológicos é a ISBSG (*Internacional Software Benchmarking Standards Group*) (AGUIAR, 2008). Vazquez, Simões e Albert (2008) afirmam, porém, que existe uma grande variação nos valores de produtividade para projetos com características tecnológicas semelhantes, devido a diversos fatores, tais como utilização de diferentes ferramentas e ambientes de desenvolvimento, ausência de metodologia de desenvolvimento, experiência da equipe, tamanho do projeto, reutilização de código, etc. Diante disso pode-se concluir que utilizar o valor de produtividade a partir de indicadores de mercado pode ser bastante arriscado na estimativa de esforço de projetos.

A maneira mais adequada de se obter indicadores de produtividade para calcular o esforço em projetos de software com pontos de função é apurar esse indicador por meio dos próprios projetos desenvolvidos na organização. Para isso é essencial que a organização possua uma base histórica dos projetos finalizados, de forma que seja possível calcular a produtividade média para todos os possíveis ambientes de desenvolvimento, contemplando o nível de experiência da equipe, linguagens de programação, etc. Essa produtividade média será utilizada como padrão para estimativas de esforço e deverá ser periodicamente recalculada, conforme novos projetos forem concluídos (VAZQUEZ, SIMÕES, ALBERT, 2008).

## 4. EXPERIMENTOS E RESULTADOS

### 4.1. Estudo de Caso

Este capítulo tem por objetivo demonstrar o processo de estimativa de esforço utilizado atualmente pela Fábrica de Software da empresa TOTVS, unidade Belo Horizonte, apresentando de forma sucinta sua metodologia de desenvolvimento de software e seu modelo empírico utilizado para estimativa de esforço dos requisitos durante a elaboração de uma especificação técnica.

Posteriormente será apresentado um exemplo prático para estimar o esforço de um projeto de customização, elaborado dentro dos padrões estabelecidos pela metodologia da empresa, e a aplicação ao mesmo projeto da técnica de pontos por função.

#### 4.1.1. Metodologia da Fábrica de Software – TOTVS BH

A metodologia de trabalho da Fábrica de Software da unidade TOTVS – Belo Horizonte baseou-se nos padrões definidos para UML (*Unified Modeling Language*), notação para modelagem de sistemas orientados a objetos, agregando também elementos do PMBOK®, versão 2000, implementado pelo PMI - *Project Management Institute*.

A metodologia foi baseada também em valores culturais da empresa, contempla apenas as fases iniciais do ciclo de vida do projeto e abrange 10 processos, que são descritos de forma macro a seguir:

- **Levantamento dos Processos a serem Customizados:** O Analista de Negócios levanta as necessidades de customização no cliente e abre uma solicitação de customização com o detalhamento da demanda, que será encaminhada ao Líder de Projetos da Fábrica de Software. A solicitação de customização é o documento formal onde deverá ser descrito o detalhamento da demanda de customização;

- **Validação Metodológica do Levantamento:** O Líder de projetos realiza uma validação da solicitação de customização, verificando a clareza do texto e sua objetividade, bem como, eventuais incoerências processuais. Caso a solicitação de customização não cumpra os pré-requisitos necessários, será encaminhada de volta para o Analista de Negócios responsável pela sua elaboração, acompanhada dos pontos indicados como não-conformidade, para que seja reavaliada e novamente encaminhada para a Fábrica de Software.
- **Designação do Analista de Customização:** Este processo irá designar o analista responsável pela elaboração da Especificação de Customização. É um passo importante, pois marca a entrada formal do projeto na área operacional da Fábrica de Software. É importante também para a atualização e acompanhamento do cronograma da equipe. O líder de projetos avalia o cronograma da equipe, bem como requisitos técnicos dos Analistas disponíveis, e indica o mais adequado para a tarefa.
- **Especificação de Customização:** Neste processo será construído o documento de Especificação de Customização (EC) pelo Analista de Customização designado, a partir da Solicitação de Customização. Este passo é a elaboração do projeto técnico, onde todos os requisitos necessários para o atendimento das necessidades do cliente serão levantados e analisados. Nesta fase será realizado o cálculo do esforço necessário para desenvolvimento das atividades descritas na especificação.
- **Validação Metodológica da Especificação:** Neste processo a especificação de customização é validada. Tem por objetivo garantir a qualidade e clareza das informações contidas na EC. Além disso, irá garantir que o modelo da EC seja prático e sempre produtivo, identificando eventuais pontos a serem aperfeiçoados. O Líder de projetos é responsável por fazer a revisão da especificação de customização, validando a clareza do texto e sua objetividade, bem como abrangência do escopo.
- **Envio para Análise Comercial:** Esta fase é importante para atualização do cronograma da equipe, liberando o Analista de Customização para

outras atividades, bem como para controle de quais Especificações foram feitas e estão no departamento Comercial da empresa. O líder de projetos assina a especificação, atualiza o cronograma e envia a mesma, em meio eletrônico e em papel, para o Departamento Comercial, que irá montar a Proposta Comercial.

- **Proposta Comercial:** Esta fase irá marcar o momento da composição dos valores a serem apresentados para o cliente. Toda Proposta Comercial deverá tomar como base a EC, que conterà os prazos e atividades necessários para a implementação do que foi solicitado. É conhecida a necessidade, por parte do Comercial, de se ajustarem, eventualmente, os valores, dependendo das necessidades dos clientes. Para tal, o Comercial deverá alterar o valor base da hora de customização, nunca alterando os prazos lá contidos.
- **Aceite:** É nesta fase que o cliente irá decidir sobre a proposta. O consultor de vendas apresenta a proposta ao cliente, que irá avaliar e negociar a proposta. Caso o cliente recuse a proposta, o processo é finalizado e o atendimento é concluído, constando o motivo da recusa (preço, prazo, etc.), caso o cliente aceite, o Departamento Comercial irá emitir um Pedido de Softwares e Serviços (PSS), que é o documento que garante comercialmente a execução das atividades previstas na especificação de customização.
- **Emissão da Ordem de Serviço:** É nesta fase que o Departamento Comercial irá gerar um documento do tipo “Pedido de Softwares e Serviços” (PSS), e encaminhá-lo ao líder de projetos, em meio eletrônico e em papel, devidamente assinado pelo consultor Comercial, bem como a Especificação de Customização com a assinatura do cliente.
- **Implementação do Projeto de Customização:** Nesta fase, acontece o início da implementação do Projeto de Customização. Aqui é importante que os cronogramas sejam atualizados e ocorra o correto arquivamento dos documentos gerados até o momento. O Líder de projetos irá alocar os analistas responsáveis pelo projeto (desenvolvedor, analista de testes, documentador, etc), e fornecer ao cliente o prazo de entrega e cronograma do projeto.

#### 4.1.2. Metodologia do Cálculo do Esforço da Fábrica de Software – TOTVS BH

O processo atual de cálculo dos prazos dos projetos de customização ocorre durante a etapa de “Elaboração da Especificação de Customização”, descrita no tópico 4.1.1 acima.

Atualmente, existe uma metodologia de trabalho para esta atividade, baseada numa matriz de tipos de requisitos e grau de dificuldade, onde para cada situação, é sugerido um número de horas previsto para a implementação de tal requisito. Esse número de horas é gerado a partir de dados históricos de projetos anteriores, já que, ao contrário das Fábricas de Softwares convencionais do mercado, que produzem softwares em diversas linguagens de programação, diferentes tipos de ambiente e tecnologia e soluções para vários tipos de segmentos, os projetos da Fábrica de Software do grupo TOTVS tem a peculiaridade de serem desenvolvidos exclusivamente para implementar customizações dos seus próprios aplicativos ERPs, a partir de uma ferramenta própria de desenvolvimento das customizações, denominada SDK de customização (*Software Development Kit*, ou Kit de Desenvolvimento de Software). Sendo assim, baseado principalmente na base histórica dos projetos e nos tipos de requisitos que podem ser implementados num projeto de customização, o modelo descrito a seguir foi criado pela Fábrica de Software para cálculo do esforço dos projetos.

Inicialmente foram levantados todos os tipos de atividades possíveis que um projeto de customização pode conter, chamados de Tipos de Requisitos, descritos na tabela 7:

<b>Tipos de Requisitos</b>
Elaboração da Especificação de Customização
Janela Delphi Win32
Janela .Net WinForm
Janela .Net WebForm
Relatório
Processo (ex: rotinas de cálculos, rotina de importação de dados)
Documentação Plano de Testes
Documentação Manual do Usuário
Testes
Implantação da customização

Tabela 7 – Tipos dos Requisitos ou Atividades de customização

Também foi criada uma segunda tabela para indicar o grau de dificuldade dos Requisitos:

<b>Grau de Dificuldade</b>
Baixo
Médio
Alto

Tabela 8 – Graus de dificuldade dos requisitos

Finalmente, foi elaborada uma planilha denominada “Matriz de Esforço”, a partir do produto cartesiano da tabela dos tipos de requisitos pela tabela de grau de dificuldade, onde cada valor foi obtido através da média histórica do esforço gasto em cada requisito e seu respectivo grau de dificuldade. Diferentemente da métrica por pontos de função, a metodologia adotada pela Fábrica de Software da TOTVS realiza o cálculo em horas, pois torna o processo mais prático, já que o tempo de resposta para elaboração da proposta ao cliente deve ser rápido, e além disso, o treinamento e acompanhamento recebido pelos desenvolvedores tornam a equipe de desenvolvimento bastante homogênea, diminuindo ao máximo divergências nos prazos executados entre um e outro desenvolvedor.

A tabela abaixo mostra a matriz de esforço atualmente praticada pela Fábrica de Software. Tal matriz é periodicamente retroalimentada, no mínimo uma vez ao ano, baseada na média histórica dos projetos executados durante um determinado período.

<b>Tipo do Requisito</b>	<b>Dificuldade Baixo (Horas)</b>	<b>Dificuldade Médio (Horas)</b>	<b>Dificuldade Alto (Horas)</b>
Janela Delphi Win32	12	24	36
Janela .Net WinForm	16	32	56
Janela .Net WebForm	40	80	120
Processo (rotinas de cálculos, execução de transações, etc)	16	32	48
Relatório	16	24	32
Documentação Plano de Testes	2	8	20
Documentação Manual do Usuário	4	12	20
Homologação	4	20	40
Implantação da customização	16	40	80

Tabela 9 – Matriz de Esforço

A partir da Matriz de Esforço apresentada na tabela 9, o Analista de Customização, após elaborar a Especificação de Customização, com os requisitos devidamente detalhados, elabora a planilha de atividades, com os prazos estimados para cada tarefa prevista na especificação. Por ser um modelo empírico e baseado em expertise, o Analista tem autonomia e liberdade para ajustar os prazos, de acordo com sua experiência adquirida em projetos anteriores. A próxima seção irá exemplificar uma situação em que o ajuste do prazo de um requisito foi necessário.

#### **4.1.3. Estudo de caso – exemplo prático**

Para avaliar a métrica utilizada pela Fábrica de Software, será descrito a seguir um exemplo real de um projeto de customização, de um cliente que possui os módulos de faturamento (compras e vendas), controle de estoque, financeiro e contábil dos aplicativos da linha RM Corpore da empresa TOTVS, no qual a metodologia foi aplicada na íntegra, tendo sido concluído e entregue dentro dos prazos estabelecidos, com avaliação positiva do cliente e sendo portanto considerado um caso de sucesso.

##### **4.1.3.1. Síntese do problema**

Foi verificada pelo cliente a necessidade de estabelecer preços diferenciados para cada embalagem dos produtos fornecidos pela empresa. Um produto vendido em sua unidade básica (KG) pode ter variados preços dependendo da embalagem escolhida.

Atualmente o cadastro de produtos do cliente possui diversos registros para cada produto, ou seja, cada embalagem é um produto, e isso tem trazido problemas porque as constantes transferências no controle de estoque tornam-se pouco práticas.

Para resolver o problema foi proposto um processo customizado pelo qual cada produto poderá ser cadastrado apenas uma vez na base de dados, e no cadastro de cada produto será possível informar as diversas embalagens relacionadas ao mesmo, com o respectivo preço. Desta forma será possível

trabalharmos com apenas uma quantidade real de estoque para um determinado produto.

**Requisitos a serem implementados:**

- Criação de uma tela customizada, chamada a partir da tela de visão de produtos, para que sejam cadastrados os tipos de embalagens de um determinado produto, e os respectivos preços. Essa informação será armazenada em uma tabela customizada.
- Criar uma tela customizada, chamada a partir da tela de item de movimento, para que o usuário informe a quantidade de cada tipo de embalagem para o produto que está sendo inserido no movimento.

**Não faz parte do escopo do projeto:**

- Elaboração ou alteração de qualquer relatório.

**4.1.3.2. Modelo de Solução**

- **Casos de Uso e Detalhamento**

O diagrama de caso de uso apresentado na Figura 2 foi modelado pelo Analista de Customização, e os casos de usos correspondentes ao projeto que será implementado serão detalhados nos tópicos seguintes.

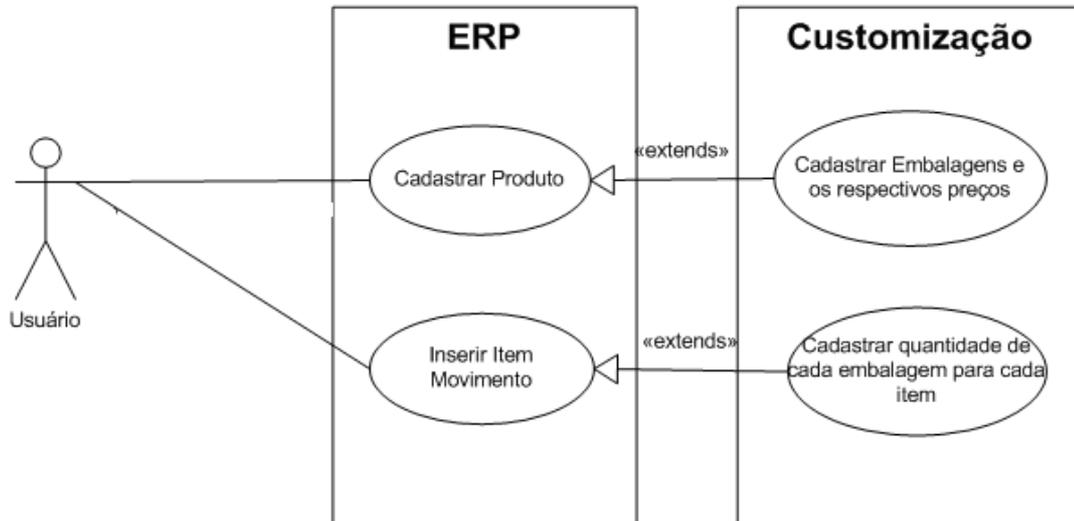


Figura 2 – Casos de Uso modelados pelo Analista de Customização a partir dos requisitos

#### 4.1.3.2.1. Processo “Cadastrar as Embalagens e os respectivos preços por produto”

- **Fluxograma do Processo (Diagrama de Atividades)**

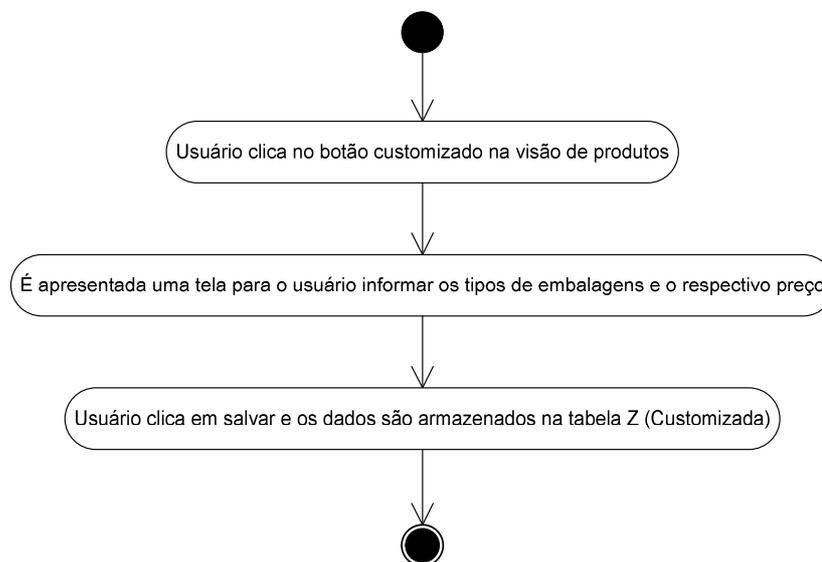


Figura 3 – Diagrama de Atividades do Caso de Uso “Cadastrar os Embalagens e os respectivos preços por produto”

- **Descrição do Processo**

1. Na tela de visão de produtos do RM Nucleus, após o usuário cadastrar um produto, ele vai selecioná-lo e clicar no botão customizado;
2. É apresentada uma tela para o usuário informar os tipos de embalagens que podem ser vendidas com o produto selecionado e o respectivo preço para cada embalagem;
3. Após informar as embalagens e preços do produto, o usuário deverá clicar em salvar e os dados serão armazenados na tabela ZPRECOPRODUTO.

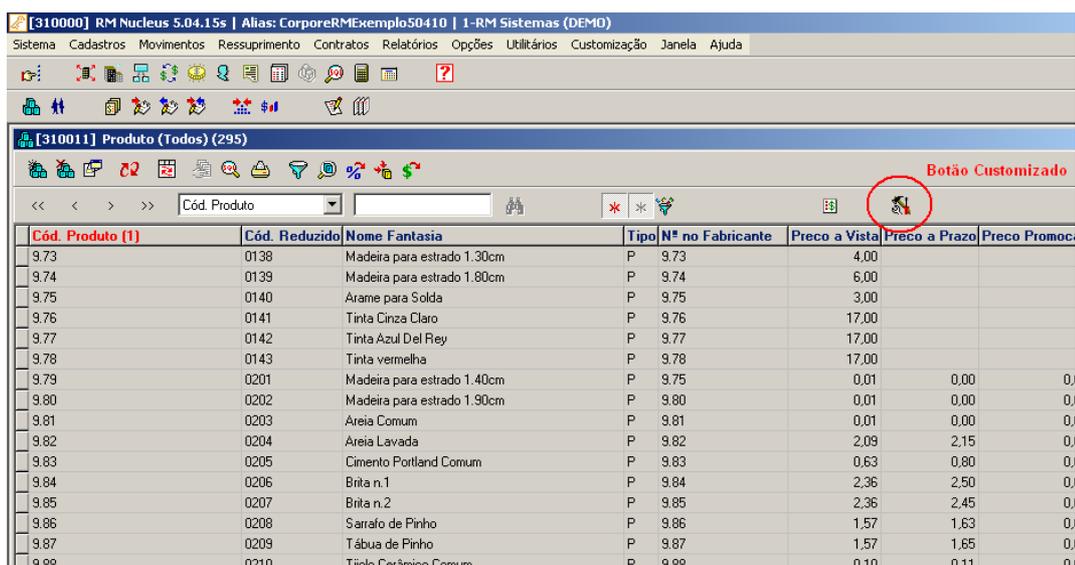
- **Detalhamento de Cálculos**

Não existem cálculos a serem detalhados.

- **Detalhamento de Processos Específicos**

Não existem processos específicos a serem detalhados.

- **Protótipos**



The screenshot shows the 'Produto (Todos) (295)' view in the RM Nucleus system. The interface includes a menu bar at the top with options like 'Sistema', 'Cadastros', 'Movimentos', etc. Below the menu is a toolbar with various icons. The main area displays a table of products with columns for 'Cód. Produto (1)', 'Cód. Reduzido', 'Nome Fantasia', 'Tipo', 'Nº no Fabricante', 'Preço a Vista', 'Preço a Prazo', and 'Preço Promocional'. A red circle highlights a button labeled 'Botão Customizado' in the top right corner of the table area.

Cód. Produto (1)	Cód. Reduzido	Nome Fantasia	Tipo	Nº no Fabricante	Preço a Vista	Preço a Prazo	Preço Promocional
9.73	0138	Madeira para estrado 1.30cm	P	9.73	4,00		
9.74	0139	Madeira para estrado 1.80cm	P	9.74	6,00		
9.75	0140	Arame para Solda	P	9.75	3,00		
9.76	0141	Tinta Cinza Claro	P	9.76	17,00		
9.77	0142	Tinta Azul Del Rey	P	9.77	17,00		
9.78	0143	Tinta vermelha	P	9.78	17,00		
9.79	0201	Madeira para estrado 1.40cm	P	9.75	0,01	0,00	0,01
9.80	0202	Madeira para estrado 1.90cm	P	9.80	0,01	0,00	0,01
9.81	0203	Areia Comum	P	9.81	0,01	0,00	0,01
9.82	0204	Areia Lavada	P	9.82	2,09	2,15	0,01
9.83	0205	Cimento Portland Comum	P	9.83	0,63	0,80	0,01
9.84	0206	Brita n.1	P	9.84	2,36	2,50	0,01
9.85	0207	Brita n.2	P	9.85	2,36	2,45	0,01
9.86	0208	Sarrafo de Pinho	P	9.86	1,57	1,63	0,01
9.87	0209	Tábua de Pinho	P	9.87	1,57	1,65	0,01
9.88	0210	Tijolo Cerâmico Comum	P	9.88	0,10	0,11	0,01

Figura 4 – Botão Customizado da Visão de Produtos

A tela apresentada na Figura 4 pertence ao sistema original e corresponde à visão dos produtos cadastrados no sistema. A tela customizada (mostrada na Figura

5) será acionada a partir do botão em destaque na Figura 4, onde o usuário deverá informar os diferentes preços por tipo de embalagem.

Nome	Múltiplo	Preço
Vasilhame 1Kg	1,00	10,00
Vasilhame 5Kg	5,00	45,00
Vasilhame 10Kg	10,00	88,00

Figura 5 – Cadastro Customizado de embalagens por produto

- **Dicionário de dados**

Tabela ZPRECOPRODUTO

Nome	Tipo	Descrição
<u>CODCOLIGADA</u>	INTEIRO	Chave seqüencial da tabela de integração.
<u>IDPRD</u>	INTEIRO	Indica o status do registro
DESCRICAOEMBALAGEM	ALFANUMERICO(40)	Indica a operação a executar. No caso dos Meios de Pagamento, a única operação possível será Inserção. Caso haja necessidade de alteração nos meios de pagamento, deverá ser feito <u>manualmente</u> a partir da tela de movimentos do RM Nucleus.
MULTIPLOQTD	INTEIRO	Múltiplo de quantidade da unidade de medida
PRECO	REAL	Preço do produto na respectiva embalagem

Tabela 10 – Estrutura da tabela ZPRECOPRODUTO

#### 4.1.3.2.2. Processo “Inserir Tipos de Embalagens X Quantidade por item de movimento”

- **Fluxograma do Processo**

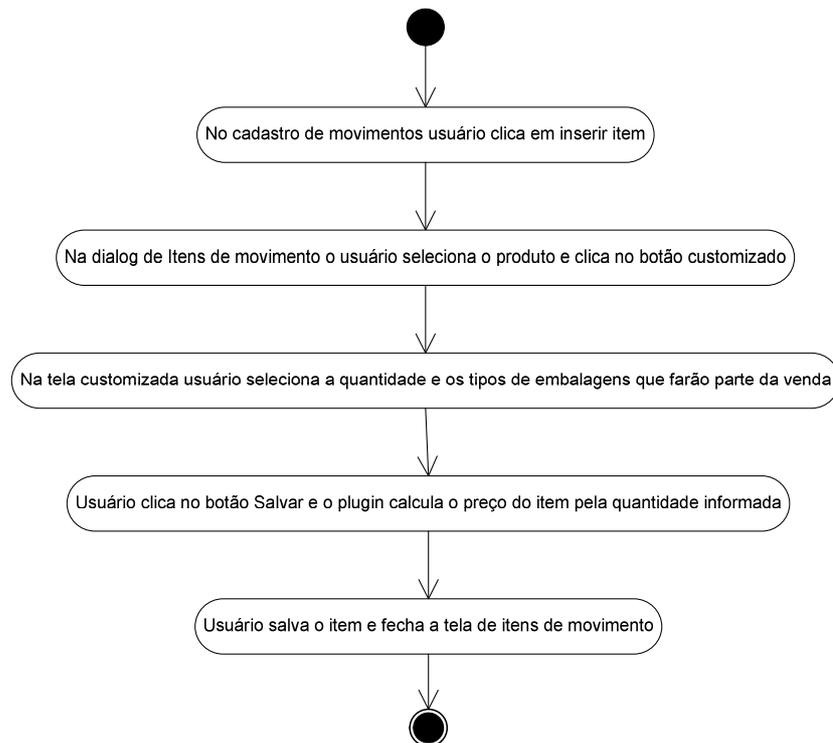


Figura 6 – Diagrama de Atividade do Caso de Uso “Inserir Tipos de Embalagens X Quantidade por item de movimento”

- **Descrição do Processo (Diagrama de Atividades)**

1. Ao cadastrar um novo movimento, o usuário clica em inserir um novo item;
2. A tela de edição de itens de movimento é mostrada, o usuário seleciona o produto e logo depois clica no botão customizado;
3. Uma nova tela customizada é acionada e o usuário deverá fazer a seleção de quais embalagens daquele produto farão parte do movimento e a quantidade de cada embalagem;
4. O usuário clica no botão salvar e o processo customizado irá somar a quantidade informada das embalagens e preencher o campo quantidade no cadastro do item, e o preço unitário será

preenchido pelo cálculo da média do valor total das embalagens pela quantidade.

5. O usuário salva o item de movimento e volta para a tela de edição de movimentos, podendo assim inserir um novo item ou finalizar o movimento.

- **Detalhamento de Cálculos**

Para que o produto apareça em um único item de movimento o processo customizado irá calcular a quantidade e o preço do item da seguinte maneira:

Suponhamos que o usuário vai cadastrar um item no movimento de venda de um determinado produto, na tela customizada ele escolhe um vasilhame de 10Kg, cujo preço é R\$88,00 e dois vasilhames de 1Kg, cujo preço de cada um é R\$10,00:

1 x Vasilhame 10Kg = R\$88,00

2 x Vasilhames 1Kg = R\$20,00

Desta forma o processo customizado irá preencher automaticamente os campos quantidade e valor total pelos valores informados na tela customizada, e o preço unitário será preenchido pela divisão do valor total pela quantidade total.

Valor Total: R\$108,00

Quantidade: 12Kg

Preço Unitário =  $108 / 12 = R\$9,00$

- **Detalhamento de Processos Específicos**

Ao informar as quantidades dos tipos de embalagens para o produto que está sendo inserido, o sistema automaticamente vai mostrar o preço total de acordo com o preço informado no cadastro customizado de embalagens de produtos, porém, caso seja necessário, o preço poderá ser modificado pelo usuário dentro do próprio movimento.

- **Protótipo**

Figura 7 – Botão Customizado no cadastro de Itens de Movimento

A tela apresentada na Figura 7 pertence ao sistema original e corresponde à tela de edição de itens de movimentos de venda. A tela customizada será acionada a partir do botão em destaque nessa figura, onde o usuário deverá informar os diferentes preços por tipo de embalagem. Os campos “Quantidade” e “Preço Unitário” da tela de edição de itens de movimento serão preenchidos automaticamente, após o usuário selecionar as quantidades das embalagens do item e clicar no botão OK, conforme protótipo apresentado na Figura 8.

Embalagem / Vasilhame	Quantidade	Valor Total
Vasilhame 10Kg	1,00	88,00
Vasilhame 1Kg	2,00	20,00

Figura 8 – Tela de Quantidade de Vasilhames por produto

#### 4.1.3.3. Elaboração da Planilha de atividades e Tempos estimados

Para elaborar os prazos, o Analista de Customização descreve as atividades do projeto organizando a tabela de atividades por fase, definidas no ciclo de desenvolvimento: Elaboração da Especificação, Implementação, Testes, Documentação e Implantação.

Na fase de implementação são adicionadas as atividades detalhadas na seção “Modelo de Solução” da Especificação de Customização. No exemplo descrito neste trabalho, as atividades do projeto são “Cadastrar as Embalagens e os respectivos preços por produto” e “Inserir Tipos de Embalagens X Quantidade por item de movimento”. Desta forma, a planilha de atividades fica composta da seguinte maneira:

<b>Atividades</b>
<b>Projeto “Preços de Produtos pela embalagem”</b>
<b>Especificação da Aplicação</b>
Levantamento das necessidades
Elaboração da Especificação detalhada
<b>Implementação</b>
Tela para informar preço por embalagem dos produtos
Tela para informar a quantidade por embalagem do produto que está sendo inserido no movimento
<b>Homologação</b>
Elaboração do Plano de Testes
Testes de Sistema
<b>Documentação</b>
Elaboração do Manual do Usuário
<b>Implantação</b>
Instalação da customização no cliente
<b>TOTAL</b>

Tabela 11 – Planilha de atividades do projeto de customização

Após ter estruturado as atividades previstas no projeto, o analista irá então aplicar a matriz de esforço apresentada na Tabela 9 da seção 4.1.2 desta monografia, informando o grau de dificuldade e o tipo de cada requisito especificado, e então os prazos são preenchidos, de acordo com os valores informados na matriz de esforço. As atividades da fase de **Especificação da Aplicação** ocorrem antes da geração da planilha de atividades, sendo assim, o Analista já tem o conhecimento

através de relatório das horas trabalhadas na respectiva fase. Neste projeto, foram gastas 8 horas em reuniões de levantamento do escopo do projeto e mais 32 horas para análise da solução e elaboração da especificação técnica. Por outro lado, as fases de **Implementação, Homologação, Documentação e Implantação** têm seus respectivos prazos calculados a partir da matriz de esforço.

Atividades	Tipo do Requisito	Grau de dificuldade	Tempo Previsto (Horas)
<b>Projeto “Preços de Produtos pela embalagem”</b>			
<b>Especificação da Aplicação</b>			
Levantamento das necessidades	Elaboração da Especificação	-	8
Elaboração da Especificação detalhada	Elaboração da Especificação	-	32
<b>Implementação</b>			
Tela para informar preço por embalagem dos produtos	Janela Win32 Delphi	Baixo	12
Tela para informar a quantidade por embalagem do produto que está sendo inserido no movimento	Janela Win32 Delphi	Baixo	12
<b>Homologação</b>			
Elaboração do Plano de Testes	Documentação Plano de Testes	Baixo	2
Testes de Sistema	Homologação	Baixo	4
<b>Documentação</b>			
Elaboração do Manual do Usuário	Documentação Manual Usuário	Baixo	4
<b>Implantação</b>			
Instalação da customização no cliente	Implantação	Baixo	16
<b>TOTAL</b>			<b>90</b>

Tabela 12 – Tabela de apropriação do esforço para cada atividade

Por ser um modelo empírico de estimativa, baseado principalmente na experiência adquirida do próprio Analista, o mesmo tem flexibilidade para ajustar os prazos, caso julgue que os valores apresentados na matriz de esforço estão divergentes do esforço médio de uma determinada atividade, devido ao grau de complexidade e experiência em projetos anteriores, semelhantes ao que está sendo especificado. Sendo assim, neste estudo de caso, o Analista optou por reduzir as horas “sugeridas” pela Matriz de esforço nas atividades de Implementação por saber que o tempo a ser gasto será bastante inferior ao que está previsto na Matriz de

Esforço, dada à simplicidade das telas a serem implementadas. O tempo previsto de Implantação foi reduzido, pois a instalação e o entendimento do produto a ser entregue são bastante simples.

Após os ajustes terem sido feitos, a planilha teve seu prazo total modificado, conforme a tabela abaixo, e então o Analista finaliza a Especificação de Customização, contemplando a descrição completa da solução a ser desenvolvida, bem como os prazos das atividades. A partir daí o fluxo é seguido, de acordo com as etapas da metodologia da Fábrica de Software, conforme apresentado no tópico 4.1.1 desta monografia.

Atividades	Tipo do Requisito	Grau de dificuldade	Tempo Previsto (Horas)
<b>Projeto “Preços de Produtos pela embalagem”</b>			
<b>Especificação da Aplicação</b>			
Levantamento das necessidades	Elaboração da Especificação	-	8
Elaboração da Especificação detalhada	Elaboração da Especificação	-	32
<b>Implementação</b>			
Tela para informar preço por embalagem dos produtos	Janela Win32 Delphi	Baixo	8*
Tela para informar a quantidade por embalagem do produto que está sendo inserido no movimento	Janela Win32 Delphi	Baixo	8*
<b>Homologação</b>			
Elaboração do Plano de Testes	Documentação Plano de Testes	Baixo	2
Testes de Sistema	Homologação	Baixo	4
<b>Documentação</b>			
Elaboração do Manual do Usuário	Documentação Manual Usuário	Baixo	4
<b>Implantação</b>			
Instalação da customização no cliente	Implantação	Baixo	4*
<b>TOTAL</b>			<b>70*</b>

Tabela 13 – Tabela com os prazos modificados empiricamente pelo analista

Na próxima seção será apresentado o cálculo de pontos de função para o mesmo projeto.

#### **4.1.3.4. Aplicando a técnica de Pontos de Função**

O tipo da contagem a ser feita será a contagem de projeto de desenvolvimento, pois se trata de um projeto novo, que será concebido a partir dos requisitos levantados.

Embora seja um projeto de customização de um sistema já existente, a estimativa deverá ser focada somente no módulo a ser desenvolvido, pois é exatamente a parte que ainda não se conhece o tamanho e esforço para implementação. Sendo assim, escopo da contagem limita-se ao tamanho do projeto que será implementado. O produto que originou a demanda de customização, ou seja, o software original que será customizado, não será considerado na contagem de pontos de função. As entidades nativas do ERP que são influenciadas pelo módulo customizado serão classificadas como Arquivos de Interface Externa.

A partir dos requisitos especificados serão identificadas as funções de dados (Arquivos Lógicos Internos e Arquivos de Interface Externa) e posteriormente as funções de transação (Entradas Externas, Saídas Externas e Consultas Externas).

O primeiro passo para a contagem das funções de dados consiste em identificar arquivos lógicos internos (ALIs). Neste caso podemos considerar o “Cadastro de preços por embalagens” um ALI, pois é um grupo de dados reconhecido pelo usuário e é mantido dentro da fronteira da aplicação.

Para saber a complexidade do ALI identificado deverão ser enumerados os registros lógicos e os itens de dados do mesmo. O cadastro de preços por embalagens será armazenado numa tabela única ZPRECOPRODUTO, que inicialmente conterà apenas 5 campos. Baseado na Tabela 1 da Seção 3.2.2.3, identificamos que o ALI – Cadastro de Preços por embalagens possui apenas um registro lógico e 5 itens de dados, sendo classificado portanto com complexidade BAIXA.

A seguir serão identificados os Arquivos de Interface Externa (AIE). Com base na especificação de customização do estudo de caso, podemos identificar dois Arquivos de Interface Externa: Cadastro de Produtos e Cadastro de Movimentos. Esses dois cadastros, reconhecidos pelo usuário, mantidos dentro da aplicação nativa, sendo apenas referenciado pela customização que será implementada, são classificados com complexidade ALTA, pois são entidades com mais de 6 registros lógicos cada, e mais de 30 itens de dados cada.

Desta forma obtemos a seguinte pontuação das funções de dados, conforme apresentado na tabela 14:

<b>Funções de Dados</b>	<b>Baixa</b>	<b>Média</b>	<b>Alta</b>	<b>PF</b>
ALI	1 (*7)	-	-	7
AIE	-	-	2 (*10)	20

Tabela 14 – Contagem das Funções de Dados

Terminada a etapa de identificação das funções de dados, serão identificadas então as funções de transações: Entradas Externas, Saídas Externas e Consultas Externas.

O processo “Cadastrar os preços por embalagens” corresponde a uma entrada externa (EE), porém, para determinar sua complexidade será necessário calcular o número de arquivos lógicos referenciados (ALR) pela EE e a quantidade de elementos de dados da EE. Neste caso, o arquivo lógico referenciado pelo cadastro dos preços por embalagens será o ALI atualizado pela EE, ou seja, o “Cadastro de preços por embalagens” e ainda o AIE lido no processamento da EE, que é o próprio “Cadastro de produtos”, possuindo portanto 2 ALR.

O número de dados referenciados para a EE “Cadastrar os preços por embalagens” será o número total de campos reconhecidos pelo usuário, ou chaves que são atualizadas pela EE. Com base na Especificação podem ser identificados 5 itens de dados referenciados, que correspondem aos campos da tabela ZPRECOVENDA, e a este número é somado um botão de exclusão que também interfere na entrada externa “cadastrar os preços por embalagens”. Por conter 2 ALR e 6 itens de dados referenciados, a EE tem complexidade MÉDIA, conforme indicado pela Tabela de Complexidade para Entradas Externas, mostrada na Tabela 3 da Seção 3.2.2.3.

Outra entrada externa identificada na aplicação será a “Inserção da Quantidade de Vasilhames por produto no item de movimento”. Esta EE possui 2 arquivos lógicos relacionados (cadastro de preços por embalagens e cadastro de itens de movimento), e possui apenas 3 itens de dados referenciados (Embalagem, quantidade e valor). Neste caso esta entrada externa terá complexidade BAIXA, conforme indicado na Tabela 3 da Seção 3.2.2.3.

Segundo a Especificação da Customização, o sistema deverá fazer um cálculo, retornando o preço unitário médio do produto, durante o cadastro dos itens de movimento. Este cálculo corresponde a uma Saída Externa, pois é um processo

elementar que envia dado para fora da fronteira da aplicação. Possui 2 arquivos lógicos relacionados: “Cadastro de preços por embalagens” e o “Cadastro de Itens de Movimento”. Os itens de dados referenciados são os campos reconhecidos pelo usuário que geram o processo elementar (campos embalagem, quantidade e valor total) e o(s) campo(s) reconhecidos pelo usuário que saem da aplicação (campos quantidade e preço unitário). Portanto esta Saída Externa possui 2 arquivos lógicos referenciados e 5 itens de dados referenciados, e, baseado na Tabela 4 da Seção 3.2.2.3, possui complexidade BAIXA.

Neste projeto não foi identificada nenhuma consulta externa. Sendo assim foram obtidas as seguintes pontuações das funções de transações:

<b>Funções de transações</b>	<b>Baixa</b>	<b>Média</b>	<b>Alta</b>	<b>PF</b>
EE	1 (*3)	1 (*4)	-	7
SE	1 (*4)	-	-	4
CE	-	-	-	0

Tabela 15 – Contagem das Funções do tipo transação

Total de pontos de função não ajustados (PFNA) = PF (Funções de dados) + PF (Funções de transações).

$$PFNA = 38$$

Segundo o processo de cálculo de pontos de função, a próxima etapa é calcular os fatores de ajustes, de acordo com as 14 características gerais de sistemas, que avaliam a funcionalidade geral da aplicação que será desenvolvida. Após avaliar o critério para cada característica, descrito no capítulo 3 desta monografia, obteve-se o seguinte resultado para os fatores de ajustes, conforme apresentado na Tabela 16 a seguir:

<b>Característica Geral</b>	<b>Grau de Influência</b>
Comunicação de dados	4
Processamento de Dados Distribuído	3
Desempenho	0
Carga de máquina	0
Volume de transações	3
Entrada de dados online	5

Eficiência do usuário final	3
Atualização online	3
Complexidade de processamento	0
Reusabilidade	1
Facilidade de Instalação	1
Facilidade de Operação	0
Múltiplos Locais	2
Facilidade de Alteração	0
<b>Grau de influência total</b>	<b>25</b>

Tabela 16 – Contagem dos níveis de influência

A partir do grau de influência total, será calculado o valor do fator de ajuste, através da fórmula  $VFA = (NIT * 0,01) + 0,65$ .

$$VFA = (25 * 0,01) + 0,65 = 0,90$$

Uma vez calculados os PF não ajustados e o fator de ajuste, é possível calcular os PFs ajustados (PFA). Para projetos de desenvolvimento, o cálculo é dado por  $PF \text{ ajustados} = PFNA * VFA$

$$PF \text{ Ajustados} = 38 * 0,9 = 34,2$$

Com base na técnica de estimativa por pontos de função, o projeto deste estudo de caso possui, portanto, 34,2 pontos de função.

#### **4.1.3.5. Cálculo do Esforço**

Como dito no capítulo 3, não existe uma produtividade pré-definida para pontos de função. O cálculo da produtividade varia de acordo com uma série de fatores que interferem no esforço para implementação de cada ponto de função.

Neste estudo de caso o cálculo do esforço será feito pela quantidade de pontos de função ajustados multiplicado pela taxa de produtividade, onde a taxa de produtividade corresponde à quantidade de horas necessárias para implementação de um ponto de função.

Como não há uma base histórica de produtividade média por pontos de função nos projetos de customização da Fábrica de Software da TOTVS, será

utilizada uma taxa inicial de produtividade de 10 Horas/Pontos de Função, obtida pelo ISBSG para linguagens de programação de 4ª geração, que é a classificação da linguagem utilizada neste estudo de caso (ISBSG *apud* VAZQUEZ, SIMÕES, ALBERT, 2008, p.42).

Baseado nessa taxa de produtividade, o cálculo do esforço em horas para implementação da customização em análise ficará da seguinte maneira:

Esforço = PF Ajustados x Produtividade, onde  $E = 34,2 * 10$ , totalizando assim 342 Horas.

Segundo Vazquez, Simões e Albert (2008), para estimar o prazo total do projeto de software, deve ser ainda calculado o esforço para as atividades de levantamento de requisitos, elaboração da especificação, testes, documentação e implantação. Normalmente este cálculo é feito adicionando-se um percentual sobre as atividades de desenvolvimento do projeto, por meio de um esforço médio empregado em cada atividade.

Como a atividade de estimativa do esforço do projeto de customização ocorre somente após o levantamento e a elaboração da especificação, serão somadas ao esforço total do projeto as horas utilizadas e apontadas pelo analista nessas duas atividades: 8 horas para a atividade de levantamento dos requisitos e 32 horas para a atividade de elaboração da Especificação de Customização.

Além disso, serão aplicados os seguintes percentuais para cada atividade do ciclo de vida do projeto, não contempladas na contagem de pontos de função do sistema, conforme a Tabela 17 a seguir:

<b>Atividades do Ciclo de Vida</b>	<b>% adicional</b>
Documentação Plano de Testes	5
Documentação Manual do Usuário	5
Testes	15
Implantação da customização	10
<b>Percentual Total</b>	<b>25</b>

Tabela 17 – Apropriação dos percentuais de esforço de acordo com a atividade do ciclo de vida

O esforço total do projeto de customização, originado a partir da contagem de pontos de função ficará portanto com um total de 501 Horas, conforme detalhado na Tabela 18:

<b>Atividades</b>	<b>Tempo Previsto (Horas)</b>
<b>Projeto “Preços de Produtos pela embalagem”</b>	
<b>Especificação da Aplicação</b>	
Levantamento das necessidades	8
Elaboração da Especificação detalhada	32
<b>Implementação</b>	342
<b>Homologação</b>	
Elaboração do Plano de Testes	17
Testes de Sistema	51
<b>Documentação</b>	
Elaboração do Manual do Usuário	17
<b>Implantação</b>	
Instalação da customização no cliente	34
<b>TOTAL</b>	<b>501</b>

Tabela 18 – Planilha de Horas do Projeto baseado em Pontos de Função

## **5. CONSIDERAÇÕES FINAIS**

### **5.1. Análise dos Resultados**

Foi mostrada por meio de um estudo de caso a utilização de um processo empírico de estimativa de software, implantado junto com uma metodologia de desenvolvimento de projetos de customização da Fábrica de Software do Grupo TOTVS, unidade Belo Horizonte. Além da demonstração da metodologia utilizada para estimativas, também foi aplicado ao mesmo projeto o processo de contagens de pontos de função, técnica amplamente utilizada e aceita no mercado de software.

Diante do que foi demonstrado no estudo de caso, podemos concluir que a técnica de estimativa por pontos de função pode ser aplicada nos projetos de customização, a partir dos requisitos a serem implementados. Porém ao utilizar uma taxa de produtividade inicial para cálculo do esforço do projeto, foi verificado que a quantidade de horas previstas para o projeto ficou sete vezes maior do que o prazo real do mesmo: pelo processo empírico utilizado na Fábrica de Software foram estimadas 70 horas, enquanto que a estimativa de esforço baseada nos pontos de função do mesmo projeto, obteve-se um valor total de 501 horas.

Para que o cálculo do esforço dos projetos de customização seja feito a partir da contagem de pontos de função, deve ser feita uma adaptação na taxa de produtividade por pontos de função para esses projetos, refinando essa taxa a cada projeto concluído, de acordo com a base histórica das horas executadas e apontadas nas atividades, até que se obtenha uma taxa de produtividade dentro da realidade dos projetos de customização.

A técnica de estimativa por pontos de função tem o objetivo de obter o tamanho do software, com uma unidade de medida padrão, bastante utilizada pelo mercado de software, onde a partir desta métrica, a empresa contratante pode cotar e licitar com diferentes fornecedores, qual o custo, prazo e tecnologia a ser utilizada no projeto. No caso dos projetos de customização nos produtos da linha RM da empresa TOTVS, a situação é o oposto de outros projetos de software, pois somente a Fábrica de Software da própria empresa detém a tecnologia que permite implementar as customizações desses produtos. Por isso, o modelo de estimativa implantado pela fábrica é mais eficiente para seus projetos, devido ao amplo

conhecimento e domínio sobre as soluções a serem implementadas, e a alta produtividade do processo de estimativa utilizado pela empresa.

Conclui-se que embora seja uma técnica bastante utilizada e possa ser aplicável aos projetos de customização, não seria recomendável implantá-la na metodologia da Fábrica de Software, pois o estudo de caso mostrou que o processo empírico utilizado pela TOTVS está bastante maduro e atende à realidade dos projetos da empresa, e não há razões para substituí-lo.

No entanto, ressalta-se que toda e qualquer metodologia não deve ser vista como um meio para enrijecer e burocratizar processos, ao contrário, deve ser uma bússola, que aponta uma direção, indicando, inclusive, caminhos alternativos. A metodologia é “viva”, pode e deve ser alterada, visando uma maior produtividade e racionalidade.

## 6. REFERÊNCIAS

AGUIAR, Maurício. A Produtividade de Projetos de Desenvolvimento. 2003. Disponível em: <[http://www.bfpug.com.br/islig-rio/Downloads/A\\_Produtividade\\_dos\\_Projetos\\_de\\_Developolvimento.pdf](http://www.bfpug.com.br/islig-rio/Downloads/A_Produtividade_dos_Projetos_de_Developolvimento.pdf)>. Acesso em 19 mar. 2009.

ANDRADE, Ednéia Leonor Pereira de. Pontos de Caso de Uso e Pontos de Função na gestão de estimativa de tamanho de projetos de software orientados a objetos. 2004. 132f. Dissertação (Mestrado em Gestão do Conhecimento e Tecnologia da Informação) – Universidade Católica de Brasília, Brasília, 2004.

DEKKERS, Carol A.. Ponto de Função e Medidas – O que é um Ponto de Função? Tradução de Mauricio Aguiar. 1999. Disponível em: <[www.bfpug.com.br/Artigos/Dekkers-PontosDeFuncaoEMedidas.htm](http://www.bfpug.com.br/Artigos/Dekkers-PontosDeFuncaoEMedidas.htm)>. Acesso em 15 mar. 2009.

HAZAN, Cláudia. SIME: Uma Ferramenta de Estimativa de Projetos de Software. SERPRO. Rio de Janeiro, 2006.

MARTINS JUNIOR, Joaquim. Como escrever trabalhos de conclusão de curso: instruções para planejar e montar, desenvolver, concluir, redigir e apresentar trabalhos monográficos e artigos. 2ª edição, Editora Vozes, Petrópolis-RJ, 2008.

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS. Pró-Reitoria de Graduação. Sistema de Bibliotecas. Padrão PUC Minas de normalização: normas da ABNT para apresentação de trabalhos científicos, teses, dissertações e monografias. Belo Horizonte, 2007. Disponível em: <<http://www.pucminas.br/biblioteca>>. Acesso em: 13 mai. 2008

Project Management Institute, Inc. A Guide to The Project Management Body of Knowledge. 3ª Edição, 2004 (edição traduzida pelo PMI/MG).

PROSPECTO Preliminar de Distribuição Pública Primária e Secundária de Ações Ordinárias de Emissão da TOTVS S.A. 2006. Disponível em <<http://siteempresas.bovespa.com.br/consbov/ArquivosExibe.asp?site=C&protocolo=81676>>. Acesso em 14 mar. de 2009.

VAZQUEZ, C.E.; SIMÕES, G.S.; ALBERT, RM. Análise de Pontos de Função: Medição, Estimativas e Gerenciamento de Projetos de Software. 8ª Edição, Editora Érica Ltda, São Paulo, 2008.