

# Estimativas de Software Utilizando Análise de Pontos de Função\*

Ubirajara Petri dos Santos<sup>1</sup>, Ana Paula Canal<sup>1</sup>

<sup>1</sup>Sistemas de Informação – Universidade Franciscana (UFN)  
Conjunto I - Rua dos Andradas, 1614 - Centro – 97010-032 - Santa Maria (RS)

bira.petri@gmail.com, apc@ufn.edu.br

**Abstract.** *This work aimed to carry out a study on estimates using function point analysis and propose the automation of the analysis process through a software. This software enables systems analysts to reduce their effort in carrying out estimates in software projects, and greater precision in defining costs and delivery schedules, maintaining a historical base of the counts carried out. For the software development project, the ICONIX methodology was used. For implementation, ASTEN Processes was used, which allows for the modeling and execution of processes using Business Process Model and Notation (BPMN).*

**Resumo.** *Este trabalho teve por objetivo fazer um estudo sobre estimativas utilizando análise de pontos de função e propor a automatização do processo de análise por meio de um software. Este software possibilita aos analistas de sistemas uma redução de esforço na execução de estimativas em projetos de softwares, e uma maior precisão na definição de custos e cronogramas de entregas, mantendo uma base histórica das contagens realizadas. Para o projeto de desenvolvimento do software foi utilizada a metodologia ICONIX. Para implementação foi utilizado o ASTEN Processos, o qual permite realizar a modelagem e execução de processos utilizando Business Process Model and Notation (BPMN).*

## 1. Introdução

O aumento no uso da tecnologia da informação nos diversos segmentos do mercado para automatização dos processos é evidente; fazendo com que exista um crescimento na demanda por novos sistemas bem como a manutenção dos sistemas existentes para adaptação as suas necessidades atuais [FERREIRA, 2011].

Nas instituições públicas a aquisição de serviços de desenvolvimento e manutenção de sistemas é uma tarefa bastante complexa, pois são diversos os fatores que podem influenciar no custo e cronogramas dos projetos. Com isso a Secretaria de Logística e Tecnologia da Informação (SLTI) do Ministério do Planejamento Orçamento e Gestão através da Instrução Normativa SLTI/MP N° 4, de 2008, recomenda que a contratação de serviços de fábrica de software deve observar as normas previstas para as licitações públicas e utilizar métricas de software no planejamento da aquisição e no gerenciamento dos contratos [Pacheco 2009]. Além disso a Portaria SLTI/MP nº 31, de 29 novembro de 2010, recomenda o uso da métrica Ponto de Função para os órgãos integrantes do Sistema de Administração dos Recursos de Tecnologia da Informação (SISP) [BRASIL 2010]. O SISP, foi instituído pelo Decreto nº 1.048 de 21 de janeiro de 1994 e atualizado pelo Decreto nº 7.579 de 11 de outubro de 2011, com o objetivo de organizar a operação, controle, supervisão e

\* Trabalho Final de Graduação do Curso de Sistemas de Informação da UFN, 2021.

coordenação dos recursos de informação e informática da administração direta, autárquica e fundacional do Poder Executivo Federal, sendo facultada às empresas públicas e às sociedades de economia mista a participação no SISP.

As estimativas de prazos e custos no desenvolvimento de softwares tem sido um problema constante nas empresas fabricantes de softwares, causando muitas vezes conflitos internos nas equipes de desenvolvimento bem como desconforto junto aos clientes, quando os prazos efetivados tornam-se superiores aos previstos inicialmente; provocando inclusive aumento de custos dos projetos afetando diretamente o orçamento das empresas. A falta de conhecimento sobre análise de pontos de função, tanto pelos analistas de negócio nas empresas de software como também nas instituições demandantes, dificulta ainda mais a execução de estimativas.

Com o uso crescente da métrica de Pontos de Função (PF) no estabelecimento de contratos de software na indústria e nos órgãos públicos do Governo Brasileiro, tem-se observado a falta de maturidade na utilização da métrica [HAZAN, 2008], e esta falta persiste até os dias atuais. Neste contexto, além da capacitação dos analistas de negócio dos órgãos contratantes em Contagem de Pontos de Função e Estimativas, faz-se necessário a criação de ferramentas que auxiliem e minimizem os esforços dos analistas na atividade de mensuração dos softwares. Segundo Ferreira (2011, p.13), “Essa ferramenta será de grande utilidade para o *benchmarking* de custo, prazo e produtividade de projetos do Governo Brasileiro”.

### **1.1. Objetivo Geral**

Desenvolver uma ferramenta informatizada para auxiliar os analistas no dimensionamento de softwares permitindo assim estimativas mais precisas quanto a custos e prazos para entregas.

### **1.2. Objetivos Específicos**

- Mapear os processos elementares dos softwares a serem estimados bem como o vínculo destes com os requisitos funcionais e não funcionais do sistema.
- Identificar o conjunto de dados que compõem o sistema a ser mensurado.
- Direcionar os analistas durante as contagens fazendo com que o processo de contagem ocorra de forma transparente e intuitiva conforme o tipo de projeto e o escopo das mesmas.
- Permitir aos gestores ter uma visão dos custos envolvidos nos projetos de software.

### **1.3. Organização Conteúdo**

Este trabalho divide-se em duas etapas sendo a primeira a realização de um levantamento bibliográfico sobre os conceitos e métodos de estimativa de software e a segunda a confecção de um modelo e criação de uma ferramenta informatizada para auxiliar os analistas no dimensionamento dos projetos de softwares bem como esforço para criação dos mesmos.

## **2. Referencial Teórico**

Para chegar ao objetivo final que é a implementação da ferramenta, é necessário o entendimento de todos os conceitos, metodologias e também uma busca histórica no que se refere a dimensionamento e estimativas de software. Sendo assim neste item são apresentadas as principais métricas utilizadas na indústria brasileira: Linhas de Código, Pontos por caso de Uso e Pontos por Função, listando suas vantagens e desvantagens. A

seguir é detalhado todo o processo de análise, estimativa de software utilizando Pontos de Função bem como as formas sugeridas na literatura para dimensionamento de Esforço, Custo e Prazos para efetivação dos projetos de software.

### **2.1. Métricas para medir software**

A métrica Linhas de Código (LOC) é a mais antiga das métricas utilizadas, e tem como vantagem a facilidade de contagem podendo inclusive ser automatizada. Porém esta métrica é muito subjetiva pois um número maior de linhas de código não representa que a funcionalidade seja mais ou menos complexa, pois depende muito da sua implementação. Além disso, tornar-se bastante complicado e subjetivo aplicar esta métrica em um processo de estimativas cujo insumo é um documento inicial de requisitos. Portanto, não é recomendado o uso da métrica LOC como unidade de medida para as estimativas de tamanho [HAZAN 2015].

A métrica Pontos por Casos de Uso (PCU) foi proposta por Gustav Karner com o propósito de estimar recursos para projetos de software orientados a objeto, modelados por meio de especificação de Casos de Uso. Esta métrica também é bastante subjetiva pois a escrita de uma especificação de caso de uso, geralmente não segue um padrão podendo ter diferentes estilos de escrita. Então, esta métrica não é recomendada para utilização como unidade de medida das estimativas de tamanho de projetos de software [HAZAN 2015].

A métrica Pontos de Função (PF), definida por Allan Albrecht em 1979 é utilizada pela indústria de software até os dias atuais, de forma crescente. Ela permite uma medida de tamanho funcional de projetos de software, considerando as funcionalidades implementadas, sob o ponto de vista do usuário. O tamanho funcional é definido como “tamanho do software derivado pela quantificação dos requisitos funcionais do usuário” [DEKKERS 2003].

A contagem de Pontos de Função independe da metodologia de desenvolvimento e da plataforma utilizada no desenvolvimento. Claudia Hazan em artigo publicado em 2015 na revista de Engenharia de Software recomenda a utilização da métrica Pontos de Função nas estimativas de tamanho de projetos de software [HAZAN 2015].

### **2.2. *International Function Point Users Group (IFPUG)***

O *International Function Point Users Group (IFPUG)*, sigla inglesa que em português significa Grupo Internacional de Usuários de Pontos de Função, criado em 1986, é responsável pela atualização das regras de Contagem de Pontos de Função, descritas no *Counting Practices Manual (CPM)*, que se encontra na versão 4.3.1, publicada em 2010 no IFPUG. O IFPUG também é responsável pelo exame de certificação de especialistas em contagem de Pontos de Função, denominada *Certified Function Point Specialist (CFPS)*.

### **2.3. Manual de Práticas de Contagem (CPM)**

O aumento do uso da técnica de contagem de pontos de função, permitiu que fosse testada a descrição original da medida e tornou necessário criar um guia para interpretar as regras originais em novos ambientes. Dando origem em (Abril 1988) a Versão 2.0 do *Counting Practices Manual* do IFPUG (CPM), Manual de Práticas de Contagem de Pontos de Função do Grupo Internacional de Usuários de Pontos de Função (IFPUG) [BRASIL 2018].

O Manual de Práticas de Contagem do IFPUG, tem por objetivo:

- Fornecer uma descrição clara e detalhada da contagem de pontos de função
- Garantir que as contagens sejam consistentes com as práticas de contagem dos filiados ao IFPUG
- Fornecer um guia para permitir a contagem de pontos de função a partir do que foi disponibilizado pelas metodologias e técnicas conhecidas
- Proporcionar um entendimento comum permitindo o uso de ferramentas automatizadas para a contagem de pontos de função.

A Versão 3.0 (Abril de 1990) do Manual de Práticas de Contagem de Pontos de Função do IFPUG foi o principal marco na evolução da medição do tamanho funcional. O Comitê de Práticas de Contagem do IFPUG fez um esforço para mudar o que era uma coleção de muitas interpretações das regras para um documento coerente, que representou uma visão consensual das regras de contagem de ponto de função.

A partir da versão 3.0 até a versão atual 4.3, o manual sofreu diversas mudanças estruturais e de conteúdo que possibilitaram esclarecer regras e conceitos já existentes bem como a inclusão de dicas e exemplos para auxiliar o entendimento. Estas evoluções permitiram a aplicação da técnica de pontos de função já nas fases iniciais do desenvolvimento dos projetos de softwares, aplicando disciplinas de engenharia da informação.

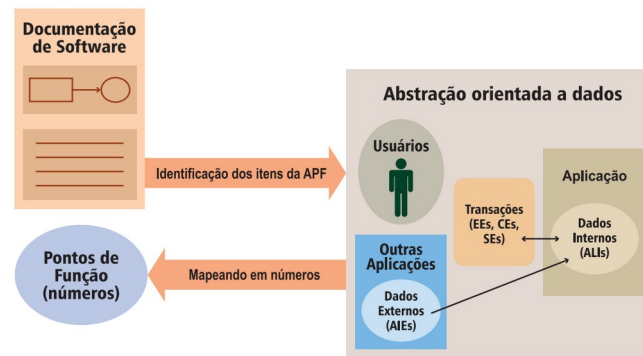
#### **2.4. Roteiro de Métricas de Software do SISP**

O Roteiro de Métricas de Software do SISP, é um Guia de Contagem de Pontos de Função proposto pelo Ministério do Planejamento, Orçamento e Gestão (MP) para a aplicação no contexto de desenvolvimento e manutenção de softwares do MP [BRASIL 2018]. Este roteiro tem como objetivo apresentar métricas, com base nas regras de contagens de pontos de função do CPM, para vários tipos de projetos de desenvolvimento e manutenção de software, promovendo o uso de métricas objetivas para contratação de serviços de desenvolvimento e manutenção de sistemas para o Governo Federal.

Além da contagem de pontos de função, este roteiro apresenta um processo de estimativas com base na métrica Ponto de Função, visando apoiar as organizações nas estimativas de tamanho, custo, prazo e esforço de seus projetos desenvolvidos internamente ou contratados.

#### **2.5. Análise de Pontos de Função**

A Figura 1 mostra uma visão macro da Análise de Pontos de Função. Segundo Vazquez (2010) a Análise de Pontos de Função (APF) é um método padrão para medição e desenvolvimento de software, e visa estabelecer uma medida do tamanho do software tomando por base a visão do usuário sobre o software e não a forma como a funcionalidade é implementada [VAZQUEZ 2010], ou seja, os requisitos do usuário é que são considerados para mensurar o software e não as tecnologias utilizadas para sua implementação. Utilizada desde meados da década de 70 para medir o tamanho funcional do software [IFPUG 2010] nesta técnica a complexidade do software é medida tomando por base o volume de arquivos lógicos mantidos e referenciados pela funcionalidade.



**Figura 1: Modelo Lógico da Análise de Pontos de Função [Roteiro de Métricas de Software do SISP 2018, p. 45]**

Os principais benefícios obtidos pelas organizações com a utilização da métrica PF são os seguintes [IFPUG 2010] [Vazquez 2010]:

- Possibilita a análise de produtividade e qualidade;
- Pode ser usada como padrão para comparação entre produtos de software;
- Apoia o gerenciamento de requisitos;
- Suporta análise de pacotes de software, considerando as necessidades da organização e as funcionalidades entregues pelo pacote;
- Pode ser usada na avaliação do porte de sistemas em Planos Diretores de Tecnologia da Informação (PDTI);
- Pode ser usada como base para o estabelecimento de contratos de software;
- Serve como insumo para as estimativas de esforço, prazo e custos

## 2.6. Contagem Estimativa de Pontos de Função

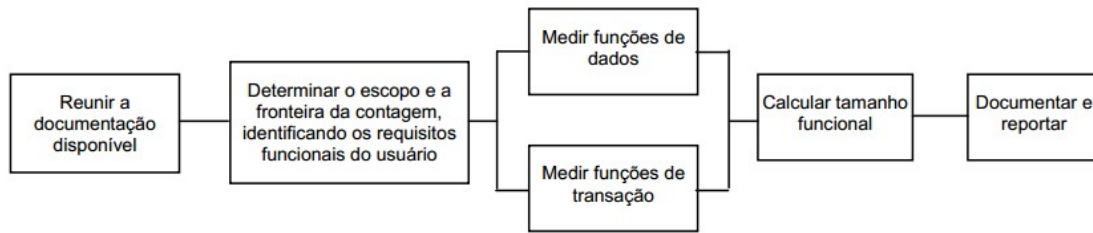
Antes de definir o método de Contagem Estimativa de Pontos de Função (CEPF), é importante destacar que “estimar significa utilizar o mínimo de tempo e esforço para se obter um valor aproximado dos Pontos de Função do projeto de software investigado” [Meli 1999 pp. 271]. Para evitar confusão, é necessária uma distinção entre os termos e conceitos: Contagem de Pontos de Função e Estimativa de Pontos de Função.

- Contagem de Pontos de Função: significa medir o tamanho do software por meio do uso das regras de contagem do IFPUG [IFPUG 2005];
- Estimativa de Pontos de Função: significa fornecer uma avaliação aproximada do tamanho de um software utilizando métodos diferentes da Contagem de Pontos de Função do IFPUG.

A ferramenta a ser implementada neste trabalho permitirá realizar tanto uma estimativa como também uma contagem final.

## 2.7. Processo de Análise

A Figura 2 mostra a sequência de procedimentos a serem executados durante a Contagem de Pontos de Função.



**Figura 2: Visão geral, gráfica, do processo de contagem de pontos de função [CPM 4.3 2010, P. 8]**

A primeira atividade a ser realizada para uma estimativa, é tomar conhecimento dos requisitos funcionais do sistema, para isso deve-se buscar informações nos documentos de visão ou lista de requisitos do sistema.

O estimador deve realizar uma leitura no documento inicial de requisitos, buscando informações relevantes para a identificação dos processos elementares. O processo elementar é definido como a menor unidade de atividade significativa para o usuário e deve ser completo em si mesmo deixando a aplicação em um estado consistente [IFPUG 2005]

Uma vez identificados os requisitos estes devem ser mapeados nos tipos funcionais da Análise de Pontos de Função (APF): Arquivo Lógico Interno (ALI), Arquivo de Interface Externa (AIE), Entrada Externa (EE), Consulta Externa (CE) e Saída Externa (SE).

A classificação da complexidade tanto dos grupos lógicos de dados como das funções transacionais é obtida identificando os dados associados a cada um. Nos casos em que não se tenha informações detalhadas referente aos dados utilizados no processo elementar, é recomendado a utilização da complexidade média para as funções transacionais e simples para as funções de dados. [Hazan 2015]. O Quadro 1 mostra a contribuição funcional dos tipos funcionais na contagem de pontos de função.

**Quadro 1: Contribuição Funcional dos Tipos Funcionais [CPM 4.3 2010, p.14 e p.21]**

Tipo Funcional	Complexidade		
	Baixa	Média	Alta
Arquivo Lógico Interno (ALI)	7 PF	10 PF	15 PF
Arquivo de Interface Externa (AIE)	5 PF	7 PF	10 PF
Entrada Externa (EE)	3 PF	4 PF	6 PF
Saída Externa (SE)	4 PF	5 PF	7 PF
Consulta Externa (CE)	3 PF	4 PF	6 PF

## 2.8. Estimativa de esforço

De posse da estimativa de tamanho, procede-se com a geração da estimativa de esforço. Hazan (2005) sugere um método simplificado de estimativa de esforço onde deve-se obter um índice de produtividade em horas/PF por meio de análise do banco de dados de histórico de projetos da organização, observando-se os atributos do projeto a ser estimado e o esforço realizado em projetos similares. Multiplicando o índice de

produtividade pela estimativa de tamanho obtemos o esforço necessário para o desenvolvimento em HH (hora\_homem).

$$\text{Esforço (Horas)} = \text{Tamanho (PF)} \times \text{Índice de Produtividade (HH/PF)}$$

## 2.9. Estimativa de prazo

As estimativas de prazo não são lineares com o tamanho do projeto. O melhor tempo de desenvolvimento onde há uma melhor relação custo x benefício de alocação de recursos e menor prazo de desenvolvimento, dado o tamanho de um projeto específico, o Roteiro do SISP sugere a utilização da fórmula de Capers Jones [JONES 2007] nas estimativas de prazo dos projetos. Esta estima o prazo, baseando-se no tamanho do projeto em pontos de função, da seguinte maneira:

$$T_d = V^t$$

Em que:

T<sub>d</sub> : Tempo de Desenvolvimento

V: Tamanho do projeto em Pontos de Função

t: Expoente definido de acordo com o Tipo de Projeto conforme Quadro 2

O prazo calculado considera todo o ciclo de vida do projeto, desde a fase de requisitos até a implantação

**Quadro 2: Expoente t por Tipo de Sistemas[ Roteiro de Métricas de Software do SISP 2018, p. 45]**

Tipo de Sistema	Expoente t
Sistema Comum – Mainframe (desenvolvimento de sistema com alto grau de reuso ou manutenção evolutiva)	0,32 a 0,33
Sistema Comum – WEB ou Cliente Servidor	0,34 a 0,35
Sistema OO (se o projeto OO não for novidade para equipe, não tiver o desenvolvimento de componentes reusáveis, considerar sistema comum)	0,36
Sistema Cliente/Servidor (com alta complexidade arquitetural e integração com outros sistemas)	0,37
Sistemas Gerenciais complexos com muitas integrações, Datawarehousing, Geoprocessamento, Workflow	0,39
Software Básico, Frameworks, Sistemas Comerciais	0,40
Software Militar (ex: Defesa do Espaço Aéreo)	0,45

## 2.10. Alocação da Equipe ao Projeto

Na alocação da equipe, deve-se considerar a estimativa de prazo e de esforço. O roteiro do SISP sugere utilizar a fórmula abaixo:

$$\text{Equipe} = \text{Esforço (HH)} / (21 \times \text{ProdDiária} \times \text{Prazo})$$

Onde:

Esforço = esforço necessário para o desenvolvimento em HH (hora\_homem)

Prazo = T<sub>d</sub> em meses

ProdDiária = 6h/dia ou 7h/dia (recomenda-se considerar 6 horas/dia)

A constante 21 equivale aos dias úteis contidos em 1 mês

O tamanho da equipe é obtido em quantidade de recursos para o desenvolvimento do projeto e deve-se considerar percentuais de alocação.

### 2.11. Estimativa de Custo

A estimativa de custo do projeto deve levar em consideração o custo de um ponto de função. Este custo deve abranger o custo da hora de todos os profissionais envolvidos no desenvolvimento da solução de software. O cálculo do custo do projeto (CP) será então da seguinte forma:

$$CP = QPF \times CPF$$

Onde:

QPF = Tamanho do projeto em PF

CPF = Custo para implementar um ponto de função na plataforma em questão.

## 3. Trabalhos Relacionados

Nesta seção, estão citados dois trabalhos publicados, sendo o primeiro um trabalho de graduação da própria UFN. Souza (2014), em seu trabalho (Desenvolvimento de Software Para Aplicação da Métrica de Estimativa Através da Análise de Pontos Por Casos de Uso), apresenta a implementação de uma ferramenta que permite registrar os casos de uso de um software e estimar a sua dimensão através do cálculo de complexidade dos atores e dos casos de uso, além disso o sistema permite a definição de valores relacionados aos fatores ambientais e técnicos. Souza deixa evidente a importância da adoção de uma ferramenta bem como os benefícios que ela pode proporcionar no processo de estimativa.

O segundo trata-se de um artigo publicado por dois autores considerados referência em grupos de discussão sobre análise de pontos de função, e também como membros de Órgãos do Governo Federal, Marinha e SERPRO, possuem experiências práticas em projetos de software. Ferreira e Hazan (2011), descrevem em seu artigo (Uma Aplicação da Análise de Pontos de Função no Planejamento e Auditoria de Custos de Projetos de Desenvolvimento de Sistemas), como a Análise de Pontos de Função pode ajudar às organizações públicas e aos órgãos de controle no planejamento, controle e na auditoria de contratos de projetos de desenvolvimento de sistemas com eficiência e eficácia.

Na conclusão do artigo os autores sugerem o desenvolvimento de uma ferramenta de estimativas e de contagem de Pontos de Função que permita o armazenamento do histórico do tamanho funcional dos projetos em Pontos de Função bem como as informações de custo, prazo e esforço. E consideram que essa ferramenta seria de grande utilidade para o *benchmarking* de custo, prazo e produtividade de projetos do Governo Brasileiro.

O trabalho de Souza (2014) deixa claro o ganho que se tem ao automatizar um processo de estimativa, trazendo agilidade e minimizando esforços por parte dos Analistas para execução desta atividade. Também fica evidente que a automatização é plenamente viável quando se tem regras e padrões bem definidos.

No artigo de Ferreira e Hazan (2011), é colocada a importância do uso de métricas principalmente no âmbito das instituições públicas onde os gestores são constantemente questionados por auditores quanto a contratos mal elaborados em serviços de software.



O artigo também evidência que a métrica Pontos de Função vai ao encontro das necessidades dos gestores, tendo em vista que a métrica dimensiona o software pela visão do usuário e não da forma como este é implementado.

Sendo assim podemos concluir que o presente trabalho vai ao encontro das necessidades do mercado de software no Brasil, podendo atender tanto Instituições Públicas como também instituições privadas. A automatização da contagem de Pontos de Função é possível haja visto que esta possui regras e processos bem definidos.

## **4. Metodologia**

Nos itens desta seção está descrito o processo de desenvolvimento bem como a ferramenta utilizada para implementação do software proposto.

### **4.1. Processo de Desenvolvimento**

A metodologia de desenvolvimento adotada para este trabalho foi o ICONIX. ROSENBERG (2005) a define como sendo um guia desde os casos de uso até a codificação e também afirma que sua missão é tirar a ambiguidade dos requisitos e proporcionar a implementação de um projeto limpo.

O ICONIX é considerado uma metodologia de desenvolvimento prática e classificada como intermediária em relação à complexidade do *Rational Unified Process* (RUP) e à simplicidade do *Extreme Programming* (XP). Sendo assim permite uma abordagem completa em cada etapa do projeto e também uma visão documentada da implementação do software.

### **4.2. Ferramenta de Desenvolvimento**

Para a implementação do Software foi utilizado o ASTEN Processos, que é uma ferramenta de desenvolvimento criada pela empresa AVMB que possibilita modelar processos, utilizando conceitos de *workflow*, árvores de decisão e *Business Process Management* (BPM)[ASTEN 2021].

A Figura 3 demonstra a arquitetura do ASTEN Processos, a qual está distribuída em três camadas distintas: Modelagem, Processamento e Execução.

A modelagem de processos é realizada em uma aplicação *Desktop*, utilizando a notação BPM, as interfaces com o usuário são definidas fazendo uso de componentes HTML e para codificação utiliza linguagem LUA. Na camada de Processamento o modelo é interpretado e processado seguindo as regras definidas na modelagem do processo. A camada de apresentação do Asten Processos foi desenvolvida utilizando a linguagem JAVA WEB, permitindo que os processos sejam executados em qualquer dispositivo utilizando Navegador web.

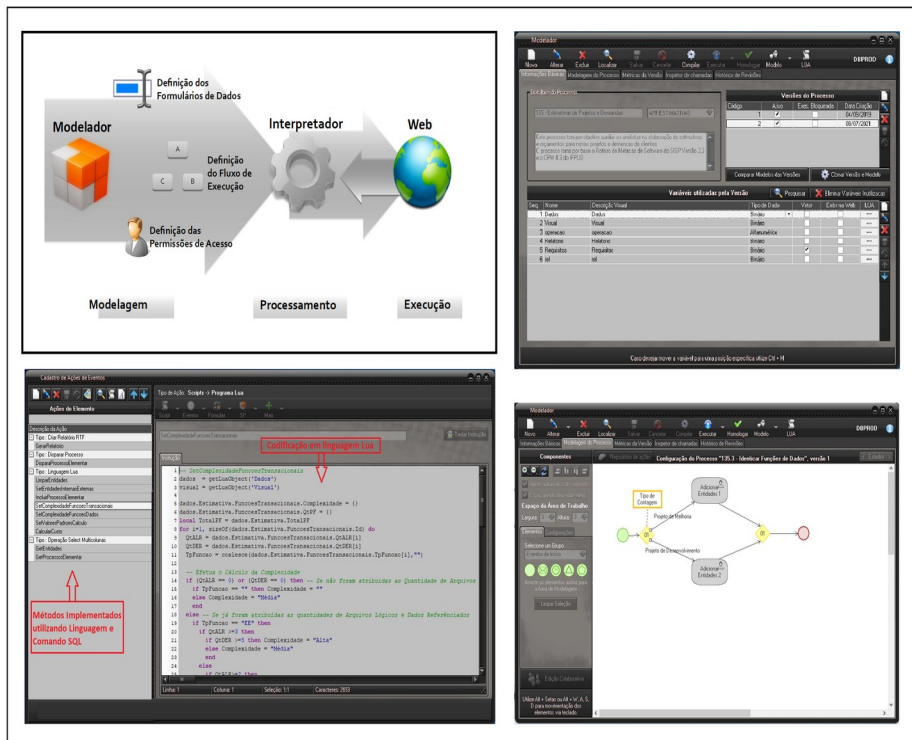


Figura 3: Arquitetura do ASTEN Processes e Janelas de Cadastro e Modelagem

## 5. Projeto e Implementação

Nesta seção estão relacionados os itens referente ao projeto e implementação do software proposto.

### 5.1. Requisitos

A Figura 4 mostra o diagrama de casos de uso onde podemos identificar os casos de usos, os atores envolvidos no processo bem como os Requisitos atendidos em cada caso de uso.

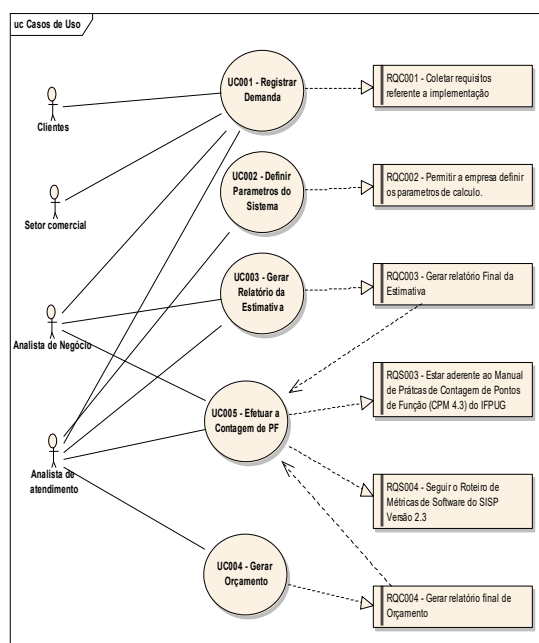


Figura 4: Diagrama de Caso de Uso

## 5.2. Diagrama de Entidades

A Figura 5 mostra o Diagrama de entidades de negócio identificadas, que compõe o software implementado. Dentre as entidades destacamos as seguintes:

**TipoProjeto:** A entidade Tipo de Projeto serve para identificar se o projeto é de desenvolvimento, melhoria, migração de dados, etc; e servirá como base para definição do fluxo a ser seguido durante o processo de estimativa.

**TipoSistema:** A entidade Tipo de Sistema, tem como objetivo classificar o Sistema à ser estimado definindo assim o expoente a ser utilizado na fórmula de Carpes Jones para definição de prazo de entrega do projeto.

**FaseProcessoDev:** A entidade Fase do Processo de Desenvolvimento conterá o percentual padrão de esforço em cada fase do processo de desenvolvimento e será utilizado nos casos em que houver mudanças de requisitos afetando assim a estimativa inicial do projeto.

**Cliente, Projeto, Sistema, Modulo, Tabela, Coluna, Linguagem, TipoContrato:** Entidades relacionadas com a parte de cadastro de projetos e sistemas a serem estimados.

**ContagemPF, VersaoContagem, FuncaoTransacional, FuncaoDados, ItemNaoMensuravel, ProcessoElementar, FatorAjuste:** São entidades relacionadas ao processo de estimativa e conterão todas as informações de cada versão de contagem realizada para um projeto.

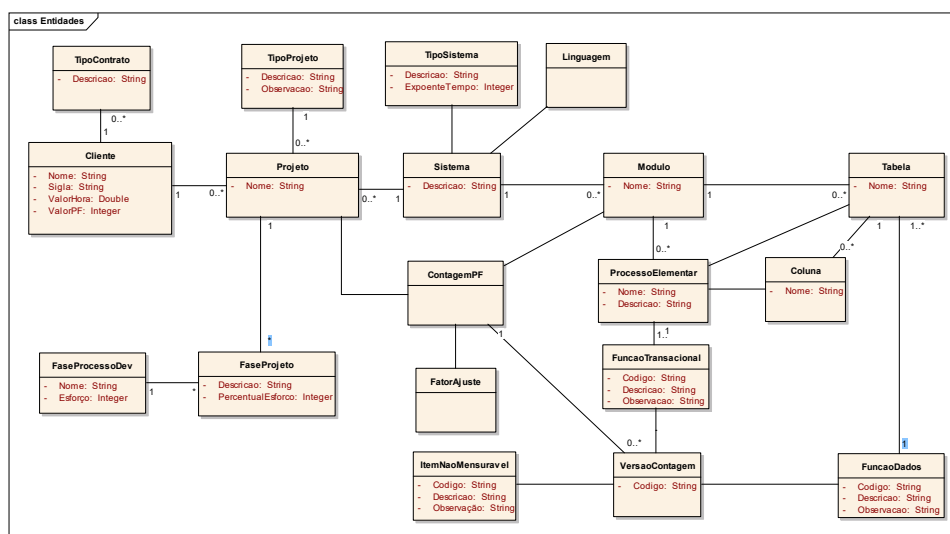
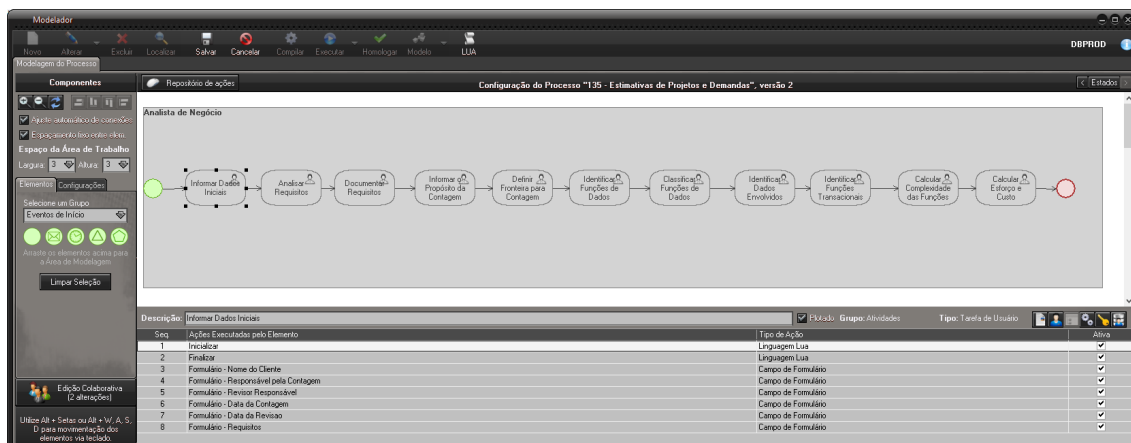


Figura 5: Diagrama de Entidades

## 5.3. Software Desenvolvido

O software desenvolvido consiste de um processo modelado no Asten Processos (Figura6), onde as atividades foram colocadas na ordem sequencial estabelecida no CPM, onde as informações são solicitadas para o usuário em uma linguagem mais natural, dispensando o uso de termos técnicos da APF.



**Figura 6: Processo Modelado no ASTEN Processes**

A primeira atividade a ser realizada é a definição do propósito da contagem onde o usuário informa o Tipo de contagem e descreve o propósito e escopo da contagem (Figura 7). A seguir o sistema começa a identificação da fronteira da contagem (Figura 8), onde é solicitado ao usuário a informação do módulo do sistema a ser medido, pois conforme orientação do Roteiro de Métricas do SISP, o módulo pode ser considerado como parâmetro para definição da fronteira da medição, facilitando assim a atividade seguinte onde são identificadas as entidades internas e externas do software medido. Para identificação das funções de dados envolvidas na medição o sistema solicita ao usuário informações de todas as entidades de dados mantidas e/ou utilizadas pelo software sendo medido (Figura 9). A seguir o usuário é solicitado a indicar dentre as entidades informadas anteriormente, quais as entidades que pertencem ao módulo informado anteriormente, com base nesta seleção o sistema classifica as entidades como sendo Arquivos Lógicos Internos (ALI) as que pertencem ao módulo e as demais como Arquivos de Interface Externa (AIE) estão fora da fronteira definida (Figura 10). O próximo passo é a identificação das funções transacionais que fazem parte do escopo da contagem; Para isso o sistema solicita ao usuário o cadastramento dos processos elementares bem como a identificação das entidades e atributos que são utilizados durante a execução do processo elementar (Figura 11).

A partir das informações coletadas nas etapas anteriores o sistema inicia a fase de identificação da complexidade das funções de dados e funções transacionais apresentando essas informações para o usuário. A seguir o sistema efetua o cálculo da quantidade de pontos de função para cada elemento da APF identificado (Figura 12).

Na etapa final o sistema solicita ao usuário os parâmetros a serem utilizados para o cálculo de Custo do projeto e a definição de prazos de entrega do software (Figura 13).

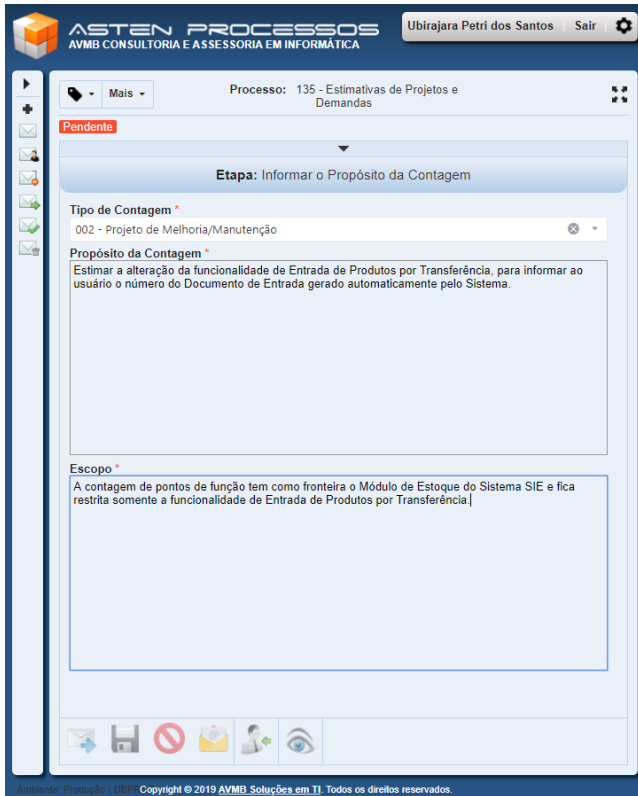


Figura 7: Propósito da Contagem

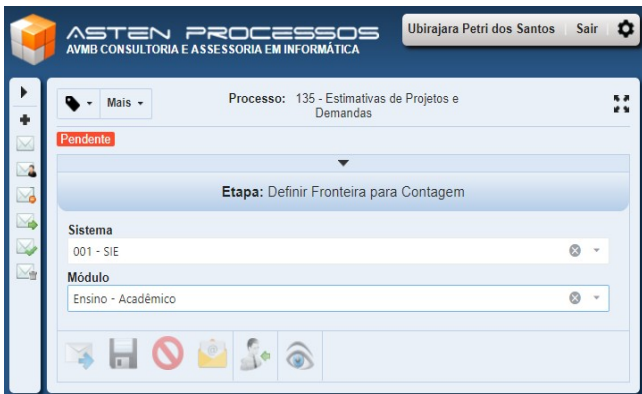


Figura 8: Definição da Fronteira da Contagem

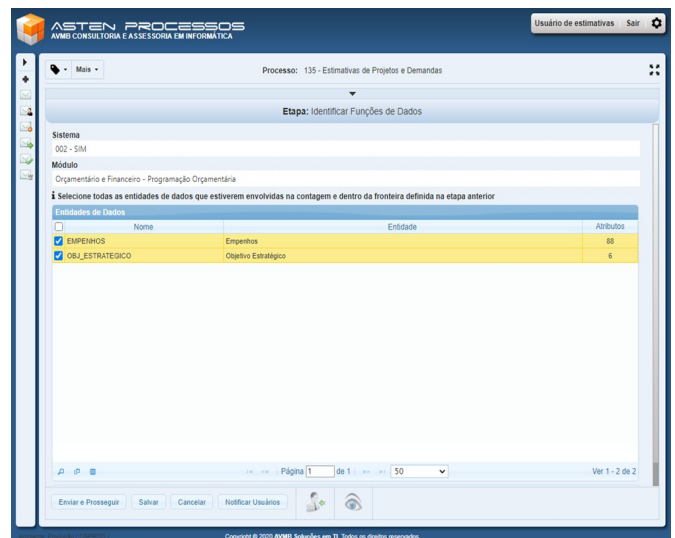


Figura 9: Interface do Sistema - Etapa 3

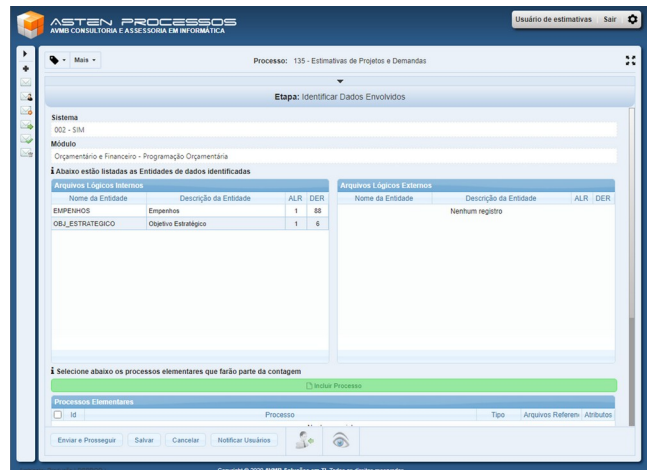


Figura 10: Identificação das Funções de Dados

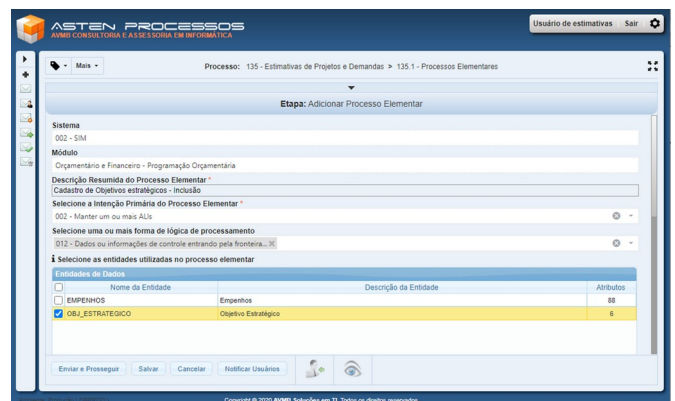


Figura 11: Cadastro de Funções Transacionais

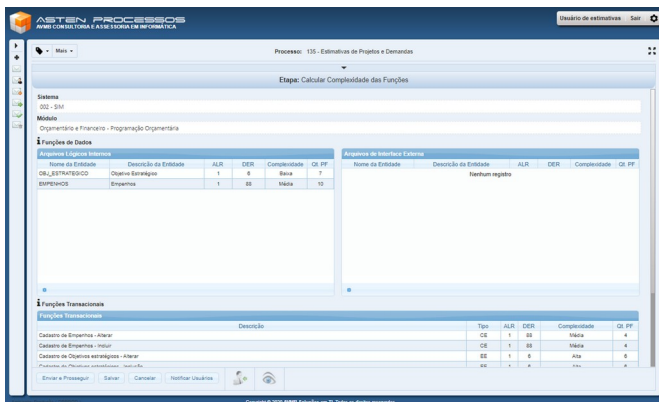


Figura 12: Cálculo da complexidade dos elementos identificados

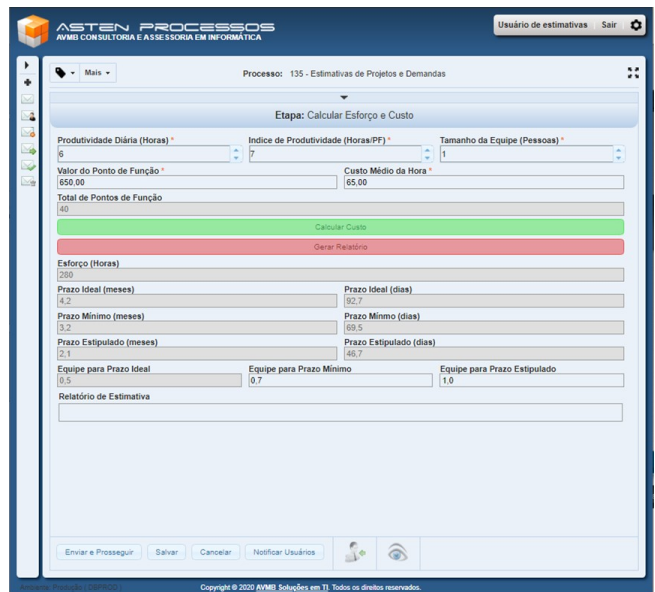


Figura 13: Cálculo do esforço e custo do projeto

Após a implementação do software, este passou por um período de validação na empresa AVMB, onde foram realizadas estimativas de pequenos projetos de manutenção evolutivas, bem como grandes projetos de desenvolvimento. Todas as estimativas foram realizadas utilizando paralelamente o software desenvolvido neste trabalho, e também da forma antiga utilizada na empresa, utilizando planilhas de cálculo.

As estimativas utilizando o software foram realizadas por analistas sem nenhum conhecimento de APF, possuindo somente conhecimento de negócio referente aos projetos e demandas solicitadas pelos clientes. As estimativas realizadas utilizando planilhas de cálculo, foram realizadas por um analista com conhecimento aprofundado em APF. Ao final as estimativas foram comparadas e apresentaram resultados muito semelhantes, o que valida a lógica implementada para identificação dos elementos da APF bem como os cálculos executados.

Conclui-se que a grande vantagem no uso do software para estimativas de pequenas demandas, é a possibilidade de entregas em um prazo menor para os clientes, pois as estimativas não precisam passar por um especialista em APF, podendo ser realizadas por uma quantidade maior de analistas, evitando assim o gargalo em um especialista em APF. Cabe salientar que as mensurações dos elementos ficam armazenadas no histórico do software, servindo como base para futuras estimativas reduzindo mais ainda o tempo de execução das mesmas.

O software também se mostrou eficaz nas estimativas realizadas em projetos maiores pois o software conduz os analistas durante o processo facilitando a identificação dos elementos da APF e evitando classificações equivocadas dos mesmos.

## 6. Conclusões

Este trabalho proporcionou colocar em prática muitos conhecimentos adquiridos durante o período acadêmico e também um aumento no conhecimento do tema Análise de Pontos de Função por parte do autor.

O Software desenvolvido atingiu o objetivo proposto e se mostrou eficaz para utilização em estimativas utilizando APF. Como citado no referencial teórico isso foi possível tendo em vista que a técnica de APF está cada vez mais consolidada, e as regras para contagem de pontos de função estão muito bem definidas no CPM pelo IFPUG, permitindo assim a automatização de diversos procedimentos do processo de análise.

Neste trabalho foram implementadas algumas funcionalidades que comprovam a importância da criação de ferramentas para auxiliar os analistas no processo de contagem de pontos de função. Uma vez concluída esta etapa, temos como objetivo explorar cada vez mais a técnica de APF e aperfeiçoar o software adicionando mais funcionalidades que facilitem o trabalho dos profissionais de TI e proporcionem uma maior produtividade nas suas entregas.

## Referências

- BRASIL, Ministério do Planejamento. Portaria SLTI/MP nº 31, de 29 novembro de 2010. Dispõe sobre recomendações técnicas para a utilização da métrica Análise de Ponto de Função no âmbito da Administração Pública Federal direta, autárquica e fundacional e dá outras providências. Brasília, DF, 2010.
- BRASIL. Ministério do Planejamento. Roteiro de Métricas de Software do SISP: versão 2.3 / Ministério do Planejamento, Desenvolvimento e Gestão. Secretaria de Tecnologia da Informação e Comunicação - Setc. – Brasília: MP, 2018.
- DEKKERS, C. Measuring the “logical” or “functional” Size of Software Projects and Software Application. Spotlight Software, ISO Bulletin May 2003 pp10-13.
- FERREIRA, R. C. C.; HAZAN, C. (2011) “Uma Aplicação da Análise de Pontos de Função no Planejamento e Auditoria de Custos de Projetos de Desenvolvimento de Sistemas”.
- HAZAN, C.; OLIVEIRA, E.A.; BLASCHEK, J.R. (2008) “How to Avoid Traps in Contracts for Software Factory Based on Function Point Metric”. 3rd International Software Measurement & Analysis Conference.
- HAZAN, C. Uma aplicação nas estimativas de tamanho de Projetos de Software, Revista Engenharia de Software 2 - Análise de pontos de função, 2015.
- IFPUG (2010) “Function Point Counting Practices Manual.(CPM)”. Release 4.3, International Function Point users Group, New Jersey.
- JONES, C. Estimating Software Costs – Bringing Realism to Estimating. 2nd Edition, McGraw Hill, New York, 2007. New York.

- MELI, R.; SANTILLO, L. Function Point Estimation Methods: A Comparative Overview. Proceedings of FESMA 99, Amsterdam, Netherlands, October 1999, pp. 271-286.
- PACHECO, A. L. F.(2009) Instrução Normativa 4/2008 SLTI/MP. Disponível em:[http://portal2.tcu.gov.br/portal/page/portal/TCU/comunidades/tecnologia\\_informacao/sefti\\_eventos/apresentacoes/2009/IN4%20-%20AeronC3%A1utica.pdf](http://portal2.tcu.gov.br/portal/page/portal/TCU/comunidades/tecnologia_informacao/sefti_eventos/apresentacoes/2009/IN4%20-%20AeronC3%A1utica.pdf), acesso em 10 Outubro 2019.
- ROSENBERG, D; STEPHENS, M; COLLINS-COPE, M. *Agile Development with ICONIX Process: People, Process, and Pragmatism*. Berkeley, US: Apress, 2005.
- VAZQUEZ, C. E.; SIMÕES, G. S.; ALBERT, R. M. (2010) Análise de Pontos de Função: Medição, Estimativas e Gerenciamento de Projetos de Software. 9ª Edição. Editora Érica, São Paulo.
- SOUZA, M. R. Desenvolvimento de Software Para Aplicação da Métrica de Estimativa Através da Análise de Pontos Por Casos de Uso, 2014.
- AVMB, Asten Processos – AVMB Soluções em TI Disponível em: <https://avmbasten.com/asten-processos/>, 2021.