

UNIVERSIDADE PAULISTA

B408FA-3 Danilo de Oliveira Dorotheu

B303GD-9 Thiago Guy Mozol Vicente

B22816-4 Marcio Fernandes Cruz

B56734-1 Diego Da Silva Santana

Aplicação de Engenharia de Requisitos em um Projeto de Software

SÃO PAULO

2015

B408FA-3 Danilo de Oliveira Dorotheu

B303GD-9 Thiago Guy Mozol Vicente

B22816-4 Marcio Fernandes Cruz

B56734-1 Diego Da Silva Santana

Aplicação de Engenharia de Requisitos em um Projeto de Software

Trabalho APS sobre aplicação de engenharia de
requisitos em um projeto de software.

Ciência da Computação 7º Semestre – Noturno

Professora Andréa Vargas.

SÃO PAULO

2015

Guia de normalização para apresentação de trabalhos acadêmicos da
Universidade Paulista : ABNT / Biblioteca Universidade Paulista, UNIP. / revisada e atualizada pelas
bibliotecárias Alice Horiuchi e Bruna Orgler Schiavi. – 2014.
49 p. : il. color.

1. NORMALIZAÇÃO. 2. TRABALHOS ACADÊMICOS. 3. ABNT. I. Biblioteca Universidade Paulista.

SUMÁRIO

1 OBJETIVO.....	5
2 INTRODUÇÃO.....	6
3 LISTA DE ILUSTRAÇÕES.....	8
4 LISTA DE TABELAS.....	9
5 CONCEITOS GERAIS.....	10
5.1 Guia SWEBOK.....	10
5.2 Engenharia de Requisitos.....	10
5.3 Requisitos funcionais.....	12
5.4 Requisitos não funcionais.....	12
5.5 Documento de requisitos de softwares.....	13
5.6 Especificação dos requisitos.....	14
6 DESCRIÇÃO DAS ATIVIDADES.....	15
6.1 Elicitação.....	15
6.1.1 Descoberta de requisitos.....	15
6.1.2 Classificação e organização dos requisitos.....	16
6.1.3 Priorização e negociação dos requisitos.....	18
6.2 Especificação.....	19
6.2.1 Requisitos de sistema.....	19
6.2.2 Requisitos funcionais.....	20
6.2.3 Requisitos não funcionais.....	20
6.2.3.1 Módulo de exportação de dados.....	20
6.2.3.2 Página de consulta web.....	21
6.3 Modelagem.....	21
6.3.1 Caso de uso do funcionamento do sistema legado da ONG.....	21
6.3.2 Caso de uso do requisito B-II.....	22
6.3.3 Caso de uso do requisito B-III.....	23
6.3.4 Requisito B-I: Criação de banco de dados remoto.....	23
6.3.5 Requisito B-IV: Exportação total dos dados.....	25
6.3.6 Requisito B-III: Tela de consulta Web.....	25
6.3.7 Requisito B-II: Exportação de dados sob demanda.....	25
6.4 Validação.....	26
6.5 Gestão.....	27
6.5.1 Planejamento de gerenciamento de requisitos.....	28
6.5.1.1 Identificação dos requisitos.....	28
6.5.1.2 Ferramentas de apoio.....	29
6.5.1.3 Gerenciamento de mudanças.....	29
7 CONCLUSÃO.....	31
8 BIBLIOGRAFIA.....	32
9 FICHA APS.....	33
Danilo.....	33
Diego.....	34
Marcio.....	35
Thiago.....	36

1 OBJETIVO

Este trabalho tem como objetivo apresentar os principais conceitos envolvendo a Engenharia de Requisitos, que é uma subárea da Engenharia de Software.

É demonstrado a aplicabilidade destes conceitos em um software fictício a ser desenvolvido para uma ONG, visando atender algumas necessidades que o sistema atual que já está em produção a duas décadas, não possui.

Foi criado o contexto em que a organização já utiliza um sistema e, que controla bem o trabalho. Este sistema é muito antigo, em modo texto, feito na linguagem *Clipper*¹ com tabelas *DBASE*². Porém, há novos requisitos que fazem necessário reescrever este software em nova plataforma, no caso Web, ou, como segunda opção, criar um outro sistema apenas para atender estes novos requisitos e, este é o contexto do trabalho apresentado.

¹ *Clipper* é uma linguagem feita para o ambiente DOS, criada em 1984. Muitas empresas que atravessaram as décadas de 1990 até hoje, possuem ainda sistemas rodando nesta linguagem.

² *DBASE* é o formato de arquivo do padrão de banco de dados *xbase*, utilizado por muitas linguagens de programação antigas, como o *Clipper*.

2 INTRODUÇÃO

O software é uma ferramenta imprescindível e estratégica para nossas vidas, seja ela pessoal ou profissional. Ele foi criado para ajudar a controlar, organizar ou facilitar a tomada decisões em praticamente todas as áreas da humanidade.

Um determinado software é criado para atuar dentro de um domínio de problema preestabelecido e, seus usuários esperam que este desempenhe muito bem seu papel.

Os requisitos é a razão de existência do software, basicamente o que é proposto a fazer.

Na área de desenvolvimento de softwares ou sistemas, é de vital importância o correto entendimento dos requisitos, a fim de que se faça produtos de alta qualidade.

Existe um conceito na Informática que se chama ciclo de vida do software que é o arco temporal que vai desde o levantamento de seus requisitos do software, passando por seu desenvolvimento, operação até seu descarte. Para que um sistema computacional desempenhe bem seu papel, deve haver qualidade e bom entendimento dos requisitos logo na primeira fase, que é o seu levantamento.

A engenharia de requisitos é uma subárea da engenharia de software que se preocupa em coletar, analisar, documentar e gerenciar as regras de negócio em todo período de existência do software.

O termo 'requisito' não é usado de forma consistente pela indústria de software. Em alguns casos, o requisito é apenas uma declaração abstrata em alto nível de um serviço que o sistema deve oferecer ou uma restrição a um sistema. No outro extremo, é uma definição detalhada e formal de uma função do sistema.
SOMMERVILLE (2011, p. 57).

Existem vários pontos de vista sobre um mesmo requisito. Os usuários finais geralmente enxergam e solicitam as necessidades do sistema de forma mais abstrata. Pode ser que, determinado requisito seja impossível ser implementado com a tecnologia atual, por exemplo. Para todos estes detalhes é de suma importância a aplicação dos processos da Engenharia de Requisitos.

Esta será responsável pela análise das informações inicialmente fornecidas

pelos clientes. Aplicará análise de viabilidade, documentará e gerenciará os requisitos e, servirá de apoio durante todo o processo de desenvolvimento, manutenção e operação do software. Ainda no descarte do software, no fim de sua vida útil, pode servir ainda como documento de entrada para ajudar a modelar novo software substituto.

Segundo Sommerville (2011), os requisitos devem ser catalogados em vários níveis de abstração, a fim de que possa atender as necessidades de todos os envolvidos no desenvolvimento e operação do software. São organizados em duas classes, sendo requisitos de usuário e de sistema.

Sobre os requisitos de usuários, Sommerville (2011, p. 58) define “São declarações, em uma linguagem natural com diagramas, de quais serviços o sistema deverá fornecer a seus usuários e as restrições com as quais este deve operar”.

Ainda Sommerville (2011, p. 58) sobre os requisitos de sistema diz “São descrições mais detalhadas das funções, serviços e restrições operacionais do sistema de software. O documento de requisitos do sistema (às vezes, chamado de especificação funcional) deve definir exatamente o que deve ser implementado”.

3 LISTA DE ILUSTRAÇÕES

Figura 1: Requisitos não funcionais - Sommerville.....	13
Figura 2: Processo de elicitação e análise de requisitos.....	15
Figura 3: Prioridade dos requisitos.....	18
Figura 4: Caso de uso do software da ONG.....	21
Figura 5: Caso de uso do requisito B-II.....	22
Figura 6: Caso de uso do requisito B-III.....	23
Figura 7: Esquema do banco de dados para consulta de doação efetivadas.....	24
Figura 8: Evolução dos requisitos.....	28

4 LISTA DE TABELAS

Tabela 1: Classificação dos requisitos.....	16
Tabela 2: Requisitos do usuário.....	17
Tabela 3: Requisitos de sistema.....	17
Tabela 4: Requisitos funcionais.....	17
Tabela 5: Requisitos não funcionais.....	18
Tabela 6: Tabela de validação de requisitos.....	26

5 CONCEITOS GERAIS

5.1 Guia SWEBOK

O IEEE³ define a Engenharia de Software como um conjunto de processos que procuram aplicar métodos da engenharia tradicional na indústria de desenvolvimento de sistemas, a fim de que se obtenha produtos de alta qualidade.

Este instituto patrocinou a criação do SWEBOK, que é o livro guia do conhecimento da engenharia de software. Segundo SWEBOK (2004), o guia em si não apresenta nenhuma ferramenta ou implementação específica. Porém demonstra a aplicação de conceitos abstratos e genéricos que podem ser aplicados de acordo com a circunstância do projeto.

O guia SWEBOK (2004) divide a Engenharia de Software em 10 áreas de conhecimento, sendo:

- Engenharia de Requisitos
- Projeto
- Construção
- Teste
- Manutenção
- Gerenciamento de Configuração
- Gerenciamento de Engenharia de Software
- Processo de Engenharia de Software
- Ferramentas e Métodos
- Qualidade

5.2 Engenharia de Requisitos

Esta área de conhecimento da Engenharia de Software procura descobrir, mapear, validar e organizar os requisitos que justificam o desenvolvimento de um software.

³ IEEE (em português, Instituto de Engenheiros Eletricistas e Eletrônicos) é uma organização sem fins lucrativos criada nos EUA no ano de 1884 que se dedica ao desenvolvimento de padrões para a indústria em geral.

Os requisitos de um sistema são as descrições do que o sistema deve fazer, os serviços que oferece e as restrições a seu funcionamento. Estes requisitos refletem as necessidades dos clientes para um sistema que serve a uma finalidade determinada, como controlar um dispositivo, colocar um pedido ou encontrar informações. O processo de descobrir, analisar, documentar e verificar esses serviços é chamado engenharia de requisitos. SOMMERVILLE (2011, p. 57).

O sucesso de um projeto de software inicia com a aplicação de boas práticas da engenharia de requisitos. Os modeladores do software devem saber interpretar as necessidades do cliente, validá-los e defini-los em 2 grupos, sendo requisitos de usuários e requisitos de sistema.

De acordo com Sommerville (2011) os requisitos de usuários são as regras que um sistema deve funcionar na visão do usuário e, podem ser expressados numa linguagem natural e através de desenhos informais ou mesmo uma linguagem padrão de modelagem, como a UML⁴. Enquanto os requisitos de sistema são as regras expressas de forma técnica de como o software deve se comportar.

Tanto os requisitos de sistema como os de usuários devem ser descritos em diferentes níveis de abstração para servir de referência a todos os envolvidos no projeto. Primeiramente um requisito deve ser validado, posteriormente mapeado sua rastreabilidade e por último ser documentado. Neste último quesito entra a descrição de um texto formal ao solicitante, bem como, num texto mais técnico ao próprio fornecedor, em especial a equipe de desenvolvimento. A indústria de software enxergou que o texto não é suficiente para descrever e mapear as funcionalidades de um sistema. Assim sendo é aconselhável para a condução de um projeto de software desenhar em diagramas suas funcionalidades, objetivando de forma mais fácil e intuitiva o que está escrito numa linguagem natural.

Um sistema é um conjunto de serviços ou funcionalidades que possuem uma ligação ou dependência num menor ou maior grau e, da mesma forma, os requisitos inicialmente levantados possuem alguma relação de dependência. Embasando neste ponto de vista, Sommerville (2011) classifica os requisitos em dois grupos, sendo requisitos funcionais e não funcionais.

4 UML é uma linguagem de modelagem de projetos.

5.3 Requisitos funcionais

São a descrição precisa de como um sistema deve funcionar. Sommerville (2011) enfatiza que este funcionamento pode ser descrito pela visão do usuário ou através de uma visão do sistema. Enquanto a primeira, é descrito de forma mais conceitual do negócio, numa linguagem natural e com diagramas de caso de uso, por exemplo. Porém, quando os requisitos funcionais são descritos em nível de sistema, descrevem especificadamente como as funções do software devem ser desenvolvidas.

Uma grande preocupação da área de Engenharia de Requisitos é fazer um levantamento completo das funcionalidades esperadas de um software, validá-los, mapeá-los e documentá-los. Isto é o alicerce de um bom software.

Em princípio, a especificação dos requisitos funcionais de um sistema deve ser completa e consistente. Completude significa que todos os serviços requeridos pelo usuário devem ser definidos. Consistência significa que os requisitos não devem ter definições contraditórias.

SOMMERVILLE (2011, p. 60)

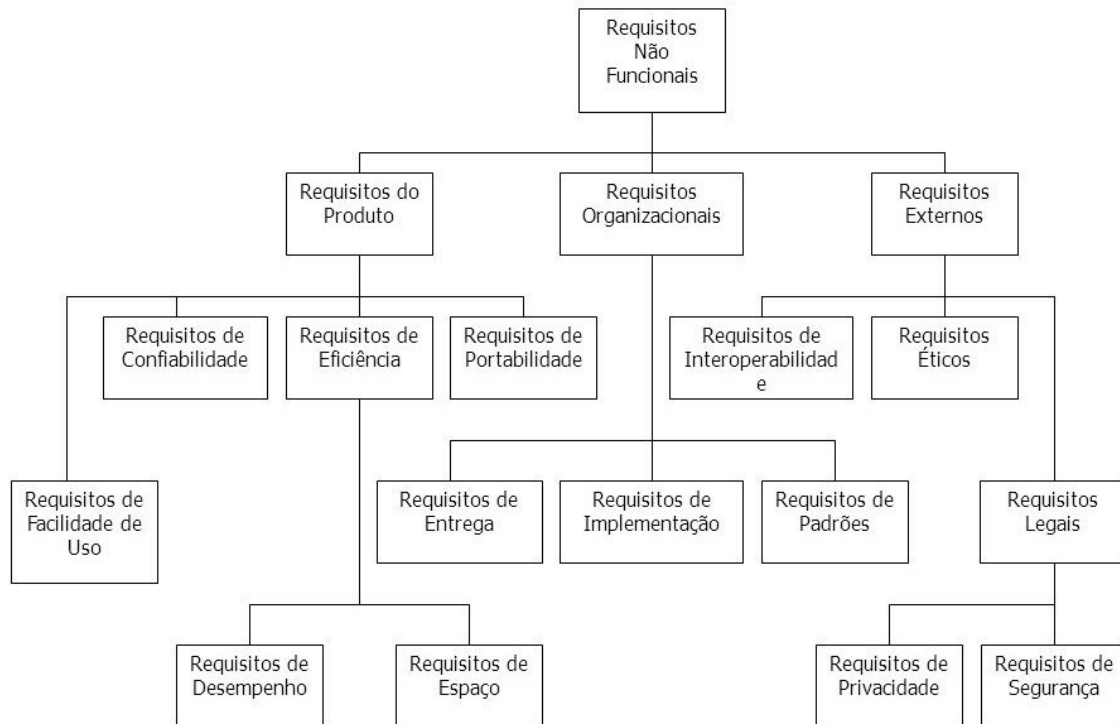
5.4 Requisitos não funcionais

Atender os serviços esperados pelo usuário e a equipe de desenvolvimento procurar seguir a risca o especificado pela modelagem não é suficiente para o sucesso de um sistema no ambiente de produção. Existem outro grupo de requisitos, definidos como não funcionais, que, se não forem atendidos, podem limitar ou barrar a operação de um software. Sommerville (2011) exemplifica características como o desempenho, disponibilidade e a proteção.

Ainda Sommerville (2011), descreve que a Engenharia de Requisitos divide os requisitos não funcionais em 3 subgrupos, sendo:

- Requisitos de produtos
- Requisitos organizacionais
- Requisitos externos

Figura 1: Requisitos não funcionais - Sommerville



Fonte: Engenharia de Software, Sommerville, 2011

5.5 Documento de requisitos de softwares

O próximo passo ao levantamento dos requisitos é levantar a estrutura dos documentos do sistema. Cada documento do sistema, segundo Sommerville (2011), é chamado *SRS – Software Requirements Specification*. O tamanho e o detalhamento da documentação depende de fatores como a complexidade do software a ser desenvolvido, da maturidade da equipe de desenvolvimento e do nível de exigência do cliente final do produto.

Boas práticas da Engenharia de Requisitos determinam que os requisitos devem ser divididos em requisitos funcionais e não funcionais, bem como, ser classificados entre requisitos de usuário e de sistema. A documentação deve procurar seguir este levantamento, porém, alguns requisitos podem ser mesclados, como por exemplo, alguns requisitos de usuários, mais abstratos, podem servir de introdução aos mesmos requisitos, expressados na forma de requisitos de sistema,

visando orientar de forma mais eficiente o time de desenvolvimento. É aceitável usar figuras, desenhos e mesmo diagramas de caso de uso para procurar expressar melhores certas necessidades.

A IEEE (1998) fornece um modelo padrão de uma documentação de requisitos que pode ser adotado ou modificado pelas fornecedoras de software. Este documento possui as seguintes sessões:

- Prefácio
- Introdução
- Glossário
- Definição dos requisitos dos usuários
- Arquitetura do sistema
- Especificação de requisitos do sistema
- Modelos dos sistemas
- Apêndices
- Índice

5.6 Especificação dos requisitos

Quando a Engenharia de Requisitos já executou o passo do levantamento e organização dos requisitos gerais do sistema e já tem o adotado o modelo de documentação, o próximo passo é a especificação dos requisitos. Este passo nada mais é do que escrever os requisitos, em um mais níveis de abstração com o objetivo de atender os usuários do sistema. Segundo Sommerville (2011), os usuários do sistema podem ser:

- Clientes do sistema
- Gerentes de Desenvolvimento
- Engenheiros de sistemas
- Engenheiros de teste de sistemas
- Engenheiros de manutenção dos sistemas

6 DESCRIÇÃO DAS ATIVIDADES

6.1 Elicitação

Depois de executado um estudo de viabilidade do software a ser desenvolvido, o projeto deve seguir para a fase de elicitação e análise. Sommerville (2011) nos demonstra o modelo que será utilizado.

Figura 2: Processo de elicitação e análise de requisitos



Fonte: Engenharia de Software, Sommerville, 2011

6.1.1 Descoberta de requisitos

Em reuniões anteriores entre os stakeholders⁵, foi demonstrado funcionamento do sistema atual da ONG. O sistema é estável e atende muito bem a organização, porém, muito antigo. É impossível executar uma evolução no sistema

⁵ Stakeholders são pessoas que tem influência direta ou indireta em algum projeto e podem influenciar o curso de sua concepção, enfim, participando diretamente na criação dos requisitos.

pois para numa barreira tecnológica. O principal requisito dos usuários é que as pessoas físicas que procedem com doações a ONG possam conseguir visualizar na Web o total do montante de dinheiro doado em determinado período. Em princípio esta informação será usada para fins de declaração de imposto de renda por parte dos doadores.

Esta necessidade foi verificada pois em todo mês de Março e Abril várias pessoas ligam para a secretaria da organização visando saber esta informação e, os usuários têm que parar suas atividades para informar verbalmente este dado, acompanhado do CNPJ e razão social da ONG para serem lançados no imposto de renda.

Os gestores da ONG apresentaram a ideia de que os próprios clientes, através de um link no portal atual da ONG, possam informar o seu CPF, alguma senha e, um período. Pode ser exibido uma lista com as doações feitas e seu total. Além de informação do CNPJ, razão social da ONG e em qual grupo deve ser declarado esta informação na Declaração Anual de Imposto de Renda.

6.1.2 Classificação e organização dos requisitos

Depois de analisado a solicitação do cliente, foi feito um levantamento de vários requisitos, e, foi definido a seguinte classificação para os requisitos, sendo:

Tabela 1: Classificação dos requisitos

Código	Grupos de requisitos
A	Requisitos de usuário
B	Requisitos de sistema
C	Requisitos funcionais
D	Requisitos não funcionais

Fonte: Especificação de engenharia de requisitos de um projeto de software, 2015

Tabela 2: Requisitos do usuário

Código	Requisito de usuário
A	Desenvolver uma página na internet onde usuário, uma pessoa física, possa consultar dados de suas doações a ONG

Fonte: Especificação de engenharia de requisitos de um projeto de software, 2015

Tabela 3: Requisitos de sistema

Código	Requisito de Sistema
B-I	Criação de banco de dados remoto.
B-II	Criação de um programa, chamado via console, para exportar dados a base de dados remota.
B-III	Criação de uma página no portal da ONG para consulta de doações efetuadas.
B-IV	Criação de um programa geral de exportação de dados.

Fonte: Especificação de engenharia de requisitos de um projeto de software, 2015

Tabela 4: Requisitos funcionais

Código	Requisito funcional
C-I	Configuração do programa de exportação de informações de doações efetuadas.
C-II	Criação de um link no portal do site e uma página que fará uma consulta aos dados armazenados.

Fonte: Especificação de engenharia de requisitos de um projeto de software, 2015

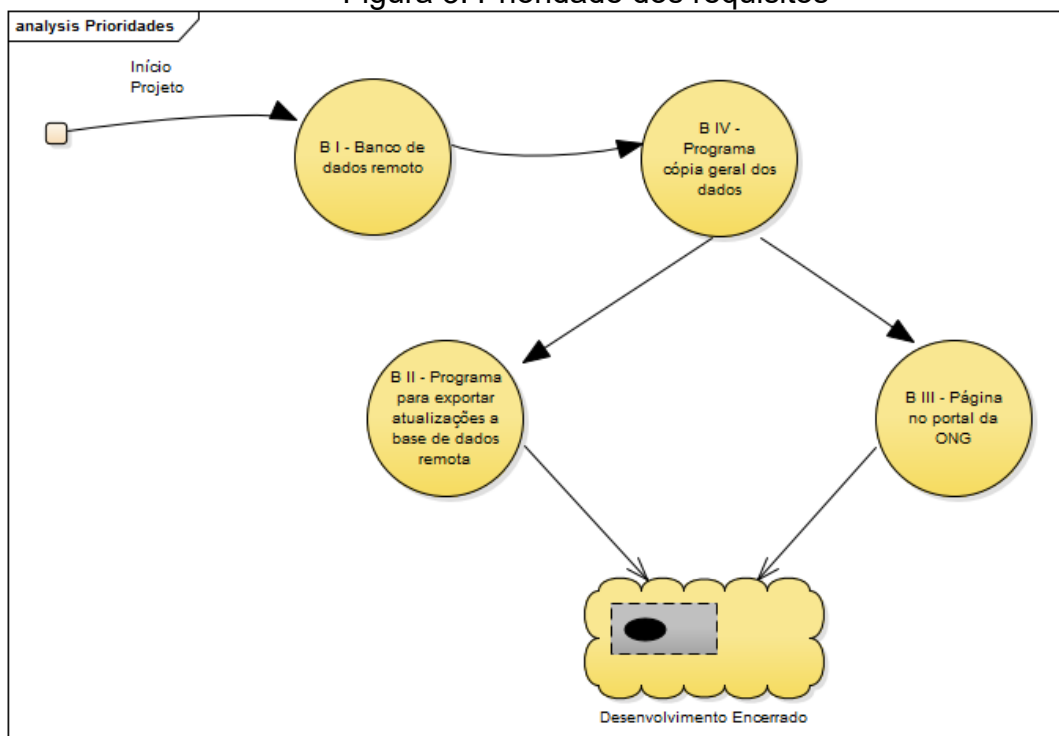
Tabela 5: Requisitos não funcionais

Código	Requisitos não funcionais
D-I	O programa de exportação de dados deve ter um bom desempenho.
D-II	O programa de exportação de dados ser registrado no log do sistema todo processo que ocorreu sucesso e que ocorreu falha.
D-III	A página web deve ser relativamente rápida o processamento.
D-IV	A página web deve ser protegida por um teste de Turing automatizado.

Fonte: Especificação de engenharia de requisitos de um projeto de software, 2015

6.1.3 Priorização e negociação dos requisitos

Figura 3: Prioridade dos requisitos



Fonte: Aplicação de eng. de requisitos em um projeto de software, 2015

Conforme a figura 3 demonstra, os requisitos devem ser desenvolvidos e entregues na seguinte ordem:

- Requisito B-I: Deve ser criado um banco de dados remoto, com o objetivo de ser espelhado as informações dos doadores e das doações em si;
- Requisito B-IV: Deve ser implementado um programa que faça toda a exportação das doações efetivadas no sistema atual;
- Requisito B-II: Este requisito pode ser desenvolvido em paralelo ao BIII. Ele representa o programa que exporta os pagamentos e estorno de doações, bem como os dados básicos do doador;
- Requisito B-III: Como explicado acima e visível na figura 3, pode ser desenvolvido em paralelo ao B-II e, representa o link e a página onde será digitado o CPF, a data de nascimento e, ao submeter, trará informações armazenadas no banco remoto.

6.2 Especificação

6.2.1 Requisitos de sistema

- Requisito B-I: Criar um banco de dados na hospedagem do site da ONG onde será mantido informações de todas as doações feitas, por cliente;
- Requisito B-II: Criar um programa para cópia de dados, que rodará diariamente, chamado pelo agendador de tarefas do computador da ONG, que exportará dados de algumas tabelas do sistema atual para o banco de dados remoto;
- Requisito B-III: Criar uma página no portal da ONG, onde o cliente informará o CPF e uma senha e, será exibido o total de doações por ano;
- Requisito B-IV: Criar um programa de exportação geral dos dados de doações efetivadas. Este será usado apenas no momento anterior a implantação do sistema em produção, ou, em último caso, na perda total dos dados do servidor remoto.

6.2.2 Requisitos funcionais

- Requisito C-I: No *deploy* do sistema será entregue um programa para alimentação da base de dados remota. É de responsabilidade do usuário a chamada do programa, que pode ocorrer vários momentos do dia. É aconselhável criar um apontamento do gerenciador de tarefas do sistema operacional, que é em Windows®;
- Requisito C-II: Em negociação paralela, foi feito uma requisição a empresa que criou o site da ONG, para que esta cria-se um link a uma chamada a uma página. Deve ser criada uma página que será colocada na hospedagem, que um usuário informará o CPF e a data de nascimento. Ao clicar em Confirmar, a página buscará no banco de dados remoto e criará uma página dinâmica, com informações de doação por ano e, no final, será exibido o CNPJ e razão social da ONG, com os dizeres: A ONG Jovens Ambientalistas declara que recebeu o total de R\$ <Valor> (valor por extenso) no ano <Ano>.

6.2.3 Requisitos não funcionais

O software será dividido em duas partes, uma de exportação e uma página remota de consulta as informações exportadas. Os requisitos não funcionais serão distintos para os dois módulos, sendo:

6.2.3.1 Módulo de exportação de dados

- O software de exportação será executado junto ao sistema legado e deve ter uma tela de configuração onde há o apontamento do caminho da conexão com a base local, como os dados de conexão da base remota;
- Deve ser relativamente rápido ao executar o processo de exportação dos dados;
- A exportação deve considerar pagamentos e estornos, ou seja, tecnicamente executará inclusões e exclusões na base de dados remota;
- Para todo o processo de exportação de dados que ocorreu sucesso, deve ser

registrado no log de eventos do sistema operacional, informando o total de transações executadas e o tempo de processamento;

- Para todo processo de exportação de dados que gerou falha, deve ser registrado no log de eventos do sistema operacional, informando o erro que o software gerou.

6.2.3.2 Página de consulta web

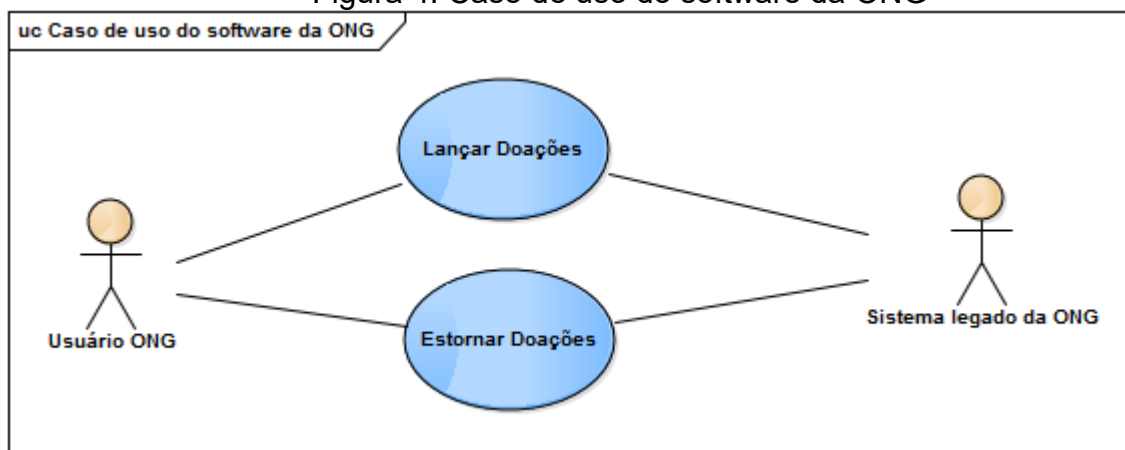
- Deve haver rapidez no processamento de dados;
- Os dados de conexão a base de dados devem estar registrados em um arquivo de configuração, junto a página;
- Toda consulta, sendo de um CPF válido ou não, deve ser registrada numa tabela de log, deve ser informado a data e hora da consulta, o IP de origem e o tempo de processamento.

6.3 Modelagem

6.3.1 Caso de uso do funcionamento do sistema legado da ONG

A ONG possui um sistema atual em funcionamento, em que os seus usuários lançam doações e estornos destas atuações numa base de dados.

Figura 4: Caso de uso do software da ONG

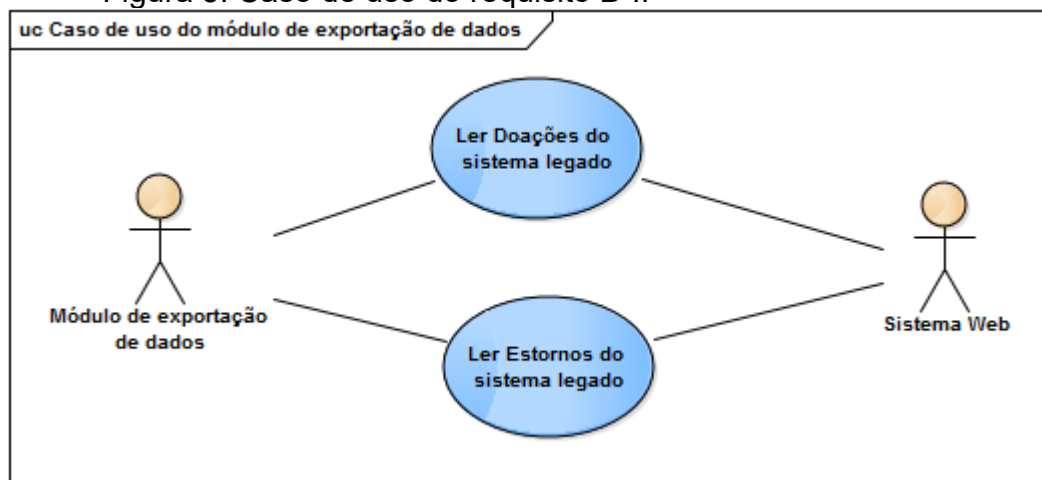


Fonte: Aplicação de eng. de requisitos em um projeto de software, 2015

6.3.2 Caso de uso do requisito B-II

Uma aplicação deve ser desenvolvida, chamada pelo agendador de tarefas do sistema operacional, fará a exportação de doações e estornos destas para uma base de dados remota.

Figura 5: Caso de uso do requisito B-II

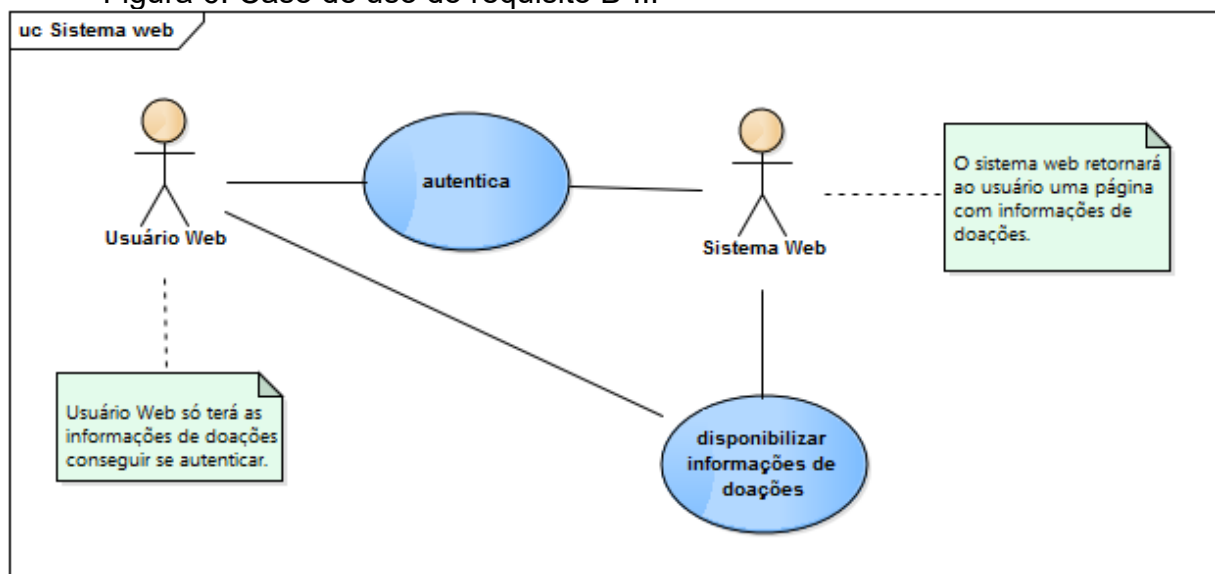


Fonte: Aplicação de eng. de requisitos em um projeto de software, 2015

6.3.3 Caso de uso do requisito B-III

Este caso de uso representa o sistema de autenticação e de disponibilização de informações de ONG. Conforme escrito na especificação, fornecerão o CPF e a data de nascimento e, o sistema web buscará as informações de pagamentos.

Figura 6: Caso de uso do requisito B-III



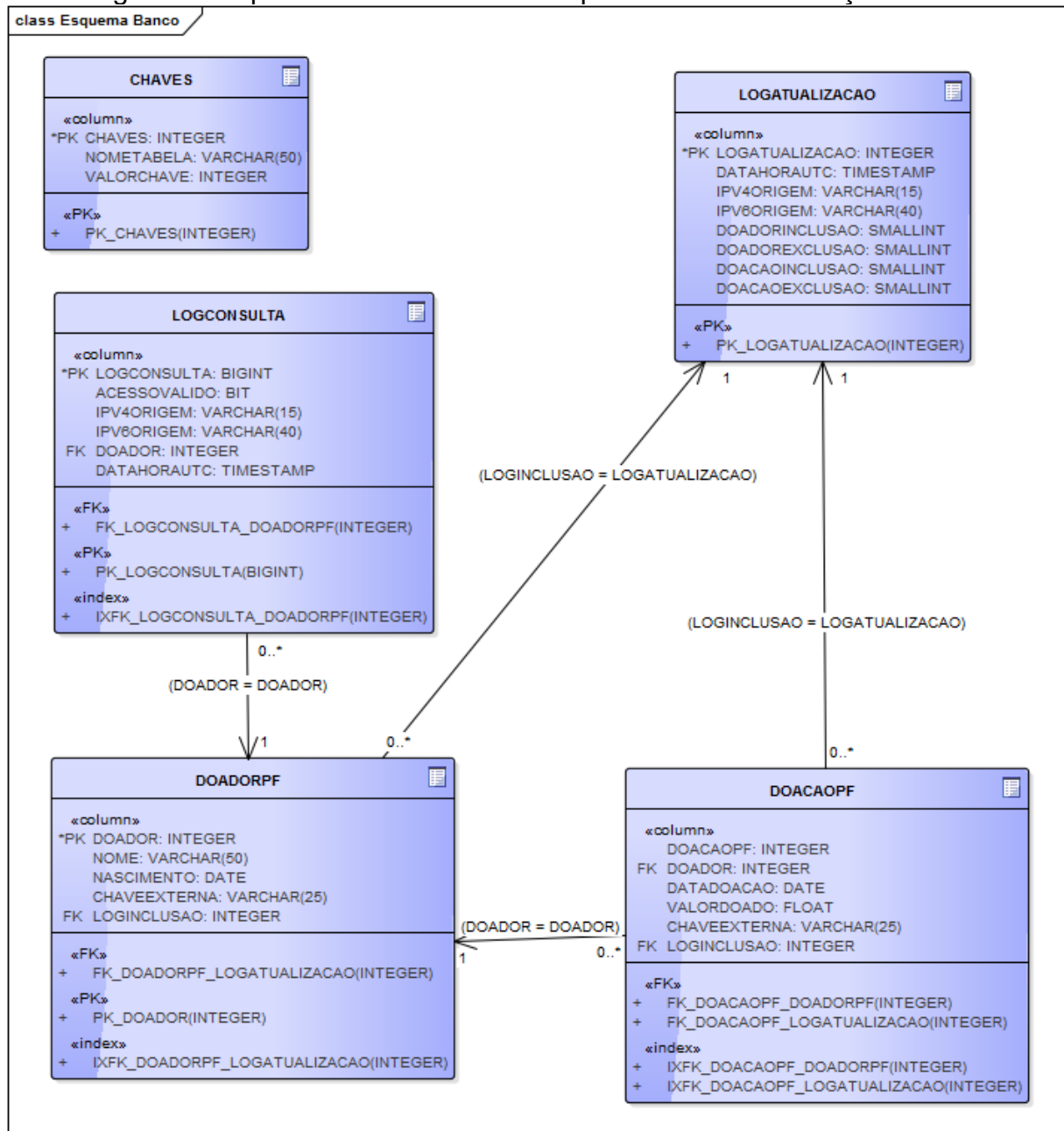
6.3.4 Requisito B-I: Criação de banco de dados remoto

Na área de hospedagem do cliente deve ser criado um novo banco de dados do fornecedor MySQL® conforme a figura 4:

- Tabela CHAVES: Esta tabela terá 4 registros, que representam o nome das 4 tabelas do negócio. Antes de incluirmos algum dado na tabela, obteremos uma próxima chave, armazenada na coluna VALORCHAVE e, só assim, procederemos com a inclusão no banco;
- Tabela LOGATUALIZACAO: Em todo processo de atualização do banco de dados, o software incluirá uma linha nesta tabela. A chave desta tabela será utilizada nas tabelas de doação para que seja possível rastrear qualquer problema de divergência de informações entre o sistema legado da empresa e seu espelho na web;

- Tabela DOADORPF: Neste objeto são persistidos os nomes dos doadores pessoas físicas que um dia, fizeram alguma doação a ONG;
- Tabela DOACAOPF: Está armazenado por data os valores doados. Em caso de estorno, a informação desta tabela será excluída;
- Tabela LOGCONSULTA: Esta tabela tem dois objetivos principais. O primeiro é saber a audiência deste projeto, se ele está sendo utilizado. O outro é para verificar possíveis tentativas de acessos indevidos. Por isso, é armazenado o IP de origem, tanto na versão 4 ou 6, bem como se o acesso é válido ou não.

Figura 7: Esquema do banco de dados para consulta de doação efetivadas



Fonte: Aplicação de eng. de requisitos em um projeto de software, 2015

6.3.5 Requisito B-IV: Exportação total dos dados

- Deve haver uma tela de configuração, amigável ao usuário final, que haverá um apontamento para a base de dados do sistema legado, bem como, um apontamento para a base de dados MySQL® remota;
- Partindo do princípio que serão todos os dados exportados, primeiramente, devem ser eliminados da base de dados remota todos os doadores e suas doações;
- Toda operação de sucesso ou de erro, seja ele de comunicação ou de exportação, deve ser gravado em eventos do sistema operacional. Assim, o usuário final poderá consultar o sucesso das operações através do Visualizador de Eventos.

6.3.6 Requisito B-III: Tela de consulta Web

- Neste ponto do projeto, como já possui dados exportados, pode ser desenvolvido a tela de consulta;
- Criar uma página na linguagem PHP (a mesma do portal), onde o usuário informará o CPF, nascimento e o Captcha⁶;
- Se o acesso for validado, informar uma tabela, quebrada por ano, informando o total doado;
- Informar o CNPJ e a razão do social da ONG, para facilitar a declaração do imposto de renda.

6.3.7 Requisito B-II: Exportação de dados sob demanda

O funcionamento deste requisito é bem parecido ao B IV. O que vai ter de

⁶Captcha é um teste de Turing automatizado, que visa comprovar que o usuário de determinado sistema é realmente um ser humano e, não um outro software. Por exemplo, um robô automatizado que tentar quebrar senhas e conseguir acesso indevido.

adicional é que este programa deverá ser chamado via console pois, será criado uma configuração para o mesmo no gerenciador de tarefas do sistema operacional. Outra diferença importante é que para saber o ponto da última atualização, deve ser verificado na tabela remota LOGATUALIZACAO a data e hora da última modificação do banco de dados e, assim, tem-se um parâmetro para saber a partir de qual ponto deve ser buscado os dados no sistema legado.

6.4 Validação

Após finalizada a análise, foi feita uma validação dos quatro requisitos de sistema a ser desenvolvido para a ONG.

Está definido o domínio de três tipos de resultado de validação, sendo:

- NA – Não aprovado
- AP – Aprovado
- Parcial

Tabela 6: Tabela de validação de requisitos

Tipo Verificação	Requisito B-I	Requisito B-II	Requisito B-III	Requisito B-IV
Validade	Parcial	Parcial	Parcial	Parcial
Consistência	AP	AP	AP	AP
Compleitude	AP	AP	AP	AP
Realismo	AP	AP	AP	NA
Verificabilidade	AP	AP	AP	NA
Revisões de requisitos	AP	AP	AP	AP
Prototipação	-	AP	-	-
Geração casos de teste	-	AP	-	-

Fonte: Especificação de engenharia de requisitos de um projeto de software, 2015

Conforme demonstrado na tabela, a equipe de *stakeholders* identificou

apenas no requisito B-IV, que é a exportação total dos dados, falta de consistência e realismo no processo e, sendo assim, devendo ser revisto. Isso ocorreu pelo fato que a base de dados do sistema legado da ONG ter mais de 20 anos. Deve ser feito uma nova versão do requisito B-IV. A exportação vai ser por período, onde o usuário informará uma data inicial e final. Como este processo será usado apenas na implantação do sistema ou em caso de perda total da base de dados remota, não vai ter nenhum agravante esta alteração da solicitação original.

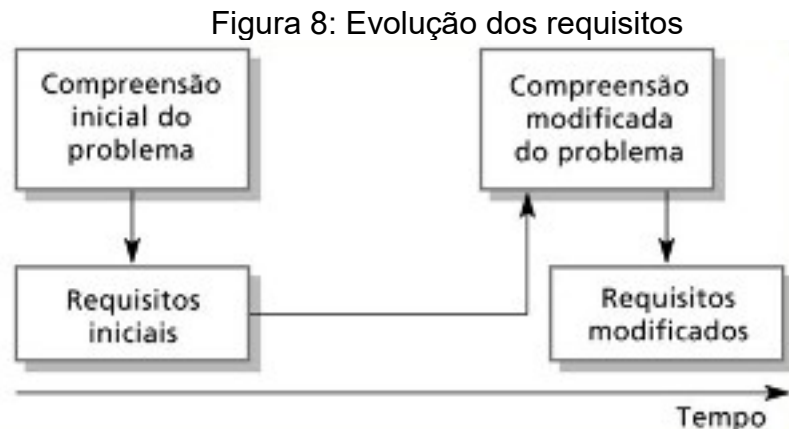
A solicitação inicial do usuário, ou seja, o requisito de usuário que deu foi que se exportasse dados de doações e de doadores do tipo pessoa física. Embora em menor número, existem doadores pessoas jurídicas, ou seja, uma empresa. Porém, esta informação veio a tona somente na fase da validação, depois de especificado a modelagem de todo o sistema, inclusive a da base de dados.

Foi feito reuniões entre os *stakeholders* e, embora o pessoal da ONG queria que na primeira versão do sistema pudesse ser consultados doações de pessoas físicas e jurídicas, foi definido que será lançado o novo sistema em produção apenas para exportação e consulta por pessoas físicas. O motivo foi o custo total do projeto, que aumentaria drasticamente e, seguindo os stakeholders da ONG, apenas 2% das doações são de empresas, e, estas, poderiam ser informadas da forma tradicional, ou seja, por contato telefônico quando solicitado.

6.5 Gestão

Segundo Sommerville (2011) os requisitos de software estão sempre mudando e, é necessário criar processos para gerenciar estas mudanças de forma metódica e correta. Embora as mudanças ocorram em sistemas maiores, qualquer software pode haver mudanças.

Uma vez que um sistema tenha sido instalado e seja usado regularmente, inevitavelmente surgirão novos requisitos. É difícil para os usuários e clientes do sistema anteciparem os efeitos que o novo sistema terá sobre seus processos de negócio e sobre a forma que o trabalho é realizado. Quando os usuários finais tiverem a experiência de um sistema, descobrirão novas necessidades e prioridades.
SOMMERVILLE (2011, p. 77).



Fonte: Engenharia de Software, Sommerville, 2011

6.5.1 Planejamento de gerenciamento de requisitos

6.5.1.1 Identificação dos requisitos

Conforme escrito no item de classificação e organização dos requisitos 16, foi dividido o projeto em 4 grupos de requisitos, sendo:

- A – Requisitos de usuários
- B – Requisitos de Sistema
- C – Requisitos funcionais
- D – Requisitos não funcionais

Na prática a codificação se guia apenas através do grupo B. Pois, segundo a Engenharia de Requisitos, os requisitos de sistema são a definição do que realmente um sistema deve fazer.

Os requisitos do grupo A, servem para gerar casos de uso e especificações com alto grau de abstração.

Quanto ao grupo C e D, orientam a equipe de desenvolvimento sobre detalhes de funcionamento do software, bem como funcionalidades não discutidas com o usuário mas, implicitamente necessárias, como disponibilidade e desempenho.

6.5.1.2 Ferramentas de apoio

Para cada projeto desenvolvido, tem-se uma instalação de um software web Wiki⁷. Como fornecedores de software, desenvolvemos uma padronização de criação e organização das páginas, que devem ser seguidas. Na página inicial, observamos os seguintes itens:

- Custeio do projeto
- Grupo A: Requisitos de usuários
- Grupo B: Requisitos de sistema
- Grupo C: Requisitos funcionais
- Grupo D: Requisitos não funcionais
- Comandos SGBD

Apenas os stakeholders da fornecedora de software possuem direito de leitura e gravação a todos os itens.

Os stakeholders da organização contratante possuem direito de leitura a todos os itens.

Os usuários finais do software tem acesso apenas ao grupo A e, neste grupo, consegue-se visualizar os diagramas de caso de uso bem como as documentações de utilização do sistema.

6.5.1.3 Gerenciamento de mudanças

As solicitações de mudanças solicitadas devem gerar novos requisitos ou, mudanças no requisito atual. Se alguma mudança solicitada se aplicar a alterações nos requisitos existentes, apenas versiona-se o requisito. Por exemplo, O requisito B-III, que inicialmente era a criação de uma página web estava na versão 1. Porém, com o advento de novas plataformas de acesso como celulares, os usuários solicitaram nova mudança, para que fosse criado um layout responsivo de acordo com o tamanho da tela. Esta solicitação de mudança necessariamente não precisa

⁷ Wiki é um conjunto de páginas interligadas. Geralmente suportada por um *framework* onde usuários podem editar, hierarquizar as páginas livremente. Possui também a característica de ter uma sintaxe acessível, onde pode-se obter apresentações visuais simples ou mais complexas, com apresentação de tabelas ou figuras.

gerar um novo requisito, basta aplicar uma nova versão ao requisito existente.

Porém, na suposição de se ter nova via de acesso, como por exemplo, um aplicativo de celular, necessariamente deveria se criar um requisito e, isto, teria reflexo nos quatro grupos de requisitos existentes (usuário, sistema, funcionais e não funcionais).

7 CONCLUSÃO

O trabalho apresentado demonstrou a aplicação dos conceitos da Engenharia de Requisitos em qualquer sistema de software. Embora projetos de softwares maiores dependem mais de padronização dos métodos propostos pela literatura, é importante também aplicar em métodos menores.

Ainda surgem discussões que burocratizar demais o levantamento e documentação inicial de requisitos podem atrapalhar o desenvolvimento do sistema como um todo. Isto leva em conta que os requisitos mudam constantemente e, embora muito documentados, sua implementação ainda gera muitos erros e retrabalho. Segundo esta visão, surgiram técnicas como metodologia ágil de desenvolvimento.

A metodologia ágil de desenvolvimento propõe um desenvolvimento de entregas incrementais e, redução ao máximo possível a documentação dos requisitos de sistema. Ainda assim, a documentação das regras de negócio bem como a documentação final do usuário permanece a mesma em relação a metodologia tradicional de desenvolvimento em cascata.

A Engenharia de Software é uma constelação de metodologias e conceitos que, baseado na Engenharia tradicional, visa resolver ou amenizar os problemas causados pela crise do software. E, o primeiro passo para o sucesso de um empreendimento, é procurar aplicar as metodologias demonstradas da engenharia de requisitos.

No software a ser desenvolvido para uma ONG, embora com poucos requisitos, desdobrou-se muitos problemas. E por isso que é importante a aplicação de métodos e processos da Engenharia de Requisitos. Com isso se aumenta a probabilidade de sucesso do software logo na implantação, bem como, deixa espaço para futura evolução e gestão eficiente. Mesmo depois de encerrado o ciclo de vida do software, os documentos levantados podem servir para a criação de um novo software, dando continuidade ao anterior. Se esta documentação não existisse, a nova equipe teria que aplicar engenharia reversa, ou mesmo, gastar tempo fazendo nova análise de algo que já existe mas, teve que evoluir.

8 BIBLIOGRAFIA

- SOMMERVILLE, Ian. Título: Engenharia de Software. Tradução Ivan Bosnic e Kalinka G. de O. Gonçalves; revisão técnica Kechi Hiramã. 9.a edição. São Paulo: Pearson Prentice Hall, 2011.
- ÁVILLA, Ana Luiza. Engenharia de Software – Introdução à Engenharia de Requisitos. Disponível em: <<http://www.devmedia.com.br/artigo-engenharia-de-software-introducao-a-engenharia-de-requisitos/8034>>. Acesso em 03 Maio. 2015.
- FALBO, Ricardo de Almeida. Engenharia de Requisitos – Notas de Aula. Disponível em: <http://www.inf.ufes.br/~falbo/files/Notas_Aula_Engenharia_Requisitos.pdf>. Acesso em 03 Maio. 2015.
- CARIDÁ, Vinicius F. ENSINO – Engenharia de Software: Engenharia de Requisitos – Parte 1. Disponível em: <<https://www.youtube.com/watch?v=wYCNjrM0lgY>>. Acesso em 03 Maio. 2015.
- CARIDÁ, Vinicius F. ENSINO – Complemento prático sobre engenharia de requisitos e modelagem. Disponível em: <https://www.youtube.com/watch?v=1_2draO54RQ>. Acesso em 03 Maio. 2015.
- VELOSO, kleber. Introdução a UML – 1.a parte. Disponível em: <<https://www.youtube.com/watch?v=hfN6n5fJfLc>>. Acesso em 03 Maio 2015.
- EDUARDO, Jean. Tutorial UML. Disponível em: <<https://www.youtube.com/watch?v=9UFvE7sWb6I>>. Acesso em 03 Maio 2015.

9 FICHA APS**Danilo****FICHA DAS ATIVIDADES PRÁTICAS SUPERVISIONADAS - APS****NOME: Danilo Dorotheu TURMA: CC6P13 RA: B408FA-3****CURSO: CIÊNCIA DA COMPUTAÇÃO CAMPUS: MARQUÊS****SEMESTRE: 6 TURNO: NOTURNO****CÓDIGO DA ATIVIDADE: Aplicação da Engenharia de Requisitos em um projeto software”****SEMESTRE: 1 ANO GRADE: 2015**

DATA DA ATIVIDADE	DESCRIÇÃO DA ATIVIDADE	TOTAL DE HORAS
17/04/15	Estudo dos itens da bibliografia	05:00:00
18/04/15	Estudo dos itens da bibliografia	05:00:00
18/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	04:00:00
19/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	06:00:00
19/04/15	Estudo UML	04:00:00
25/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	04:00:00
26/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	06:00:00
01/05/15	Estudo dos conceitos globais da Engenharia de Requisitos	03:00:00
01/05/15	Estudo UML	05:00:00
03/05/15	Escrita dos conceitos gerais	03:00:00
08/05/15	Escrita dos conceitos gerais	03:00:00
09/05/15	Escrita dos conceitos gerais	03:00:00
10/05/15	Escrita da descrição de atividades	03:00:00
16/05/15	Escrita da descrição de atividades	02:00:00
17/05/15	Escrita da descrição de atividades	05:00:00

TOTAL DE HORAS ATRIBUÍDAS: 61:00:00

Diego

FICHA DAS ATIVIDADES PRÁTICAS SUPERVISIONADAS - APS

NOME: Diego Santana TURMA: CC6P13 RA: B56734-1

CURSO: CIÊNCIA DA COMPUTAÇÃO CAMPUS: MARQUÊS

SEMESTRE: 6 TURNO: NOTURNO

CÓDIGO DA ATIVIDADE: Aplicação da Engenharia de Requisitos em um projeto software"

SEMESTRE: 1 ANO GRADE: 2015

DATA DA ATIVIDADE	DESCRIÇÃO DA ATIVIDADE	TOTAL DE HORAS
11/04/15	Estudo dos itens da bibliografia	04:00:00
18/04/15	Estudo dos itens da bibliografia	04:00:00
18/04/15	Estudo dos itens da bibliografia	04:00:00
19/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	04:00:00
19/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	04:00:00
25/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	04:00:00
26/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	04:00:00
01/05/15	Estudo dos conceitos globais da Engenharia de Requisitos	03:00:00
02/05/15	Estudo UML	08:00:00
03/05/15	Escrita dos conceitos gerais	03:00:00
08/05/15	Escrita dos conceitos gerais	03:00:00
09/05/15	Escrita dos conceitos gerais	03:00:00
10/05/15	Escrita dos conceitos gerais	03:00:00
16/05/15	Escrita dos conceitos gerais	02:00:00
17/05/15	Escrita da descrição de atividades	05:00:00

TOTAL DE HORAS ATRIBUÍDAS: 58:00:00

Marcio

FICHA DAS ATIVIDADES PRÁTICAS SUPERVISIONADAS - APS

NOME: MARCIO FERNANDES CRUZ TURMA: CC6P13 RA: B22816-4

CURSO: CIÊNCIA DA COMPUTAÇÃO CAMPUS: MARQUÊS

SEMESTRE: 6 TURNO: NOTURNO

CÓDIGO DA ATIVIDADE: Aplicação da Engenharia de Requisitos em um projeto software"

SEMESTRE: 1 ANO GRADE: 2015

DATA DA ATIVIDADE	DESCRIÇÃO DA ATIVIDADE	TOTAL DE HORAS
11/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	04:00:00
18/04/15	Estudo UML para criação de casos de uso	03:00:00
18/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	03:00:00
19/04/15	Escrita do Objetivo e Introdução do Trabalho	05:00:00
19/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	03:00:00
25/04/15	Estudo UML para criação de casos de uso	01:00:00
26/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	02:00:00
01/05/15	Estudo dos conceitos globais da Engenharia de Requisitos	04:00:00
02/05/15	Escrita dos conceitos gerais	05:00:00
03/05/15	Leitura livro Engenharia de Software, Sommerville (2011)	04:00:00
08/05/15	Escrita dos conceitos gerais	03:00:00
09/05/15	Escrita dos conceitos gerais	03:00:00
10/05/15	Escrita do Objetivo e Introdução do Trabalho	04:00:00
16/05/15	Leitura livro Engenharia de Software, Sommerville (2011)	04:00:00
17/05/15	Escrita da descrição de atividades	06:00:00

TOTAL DE HORAS ATRIBUÍDAS: 54:00:00

Thiago

NOME: THIAGO GUY TURMA: CC7P13 RA: B303GD-9

CURSO: CIÊNCIA DA COMPUTAÇÃO CAMPUS: MARQUÊS

SEMESTRE: 6 TURNO: NOTURNO

CÓDIGO DA ATIVIDADE: Aplicação da Engenharia de Requisitos em um projeto software”

SEMESTRE: 1 ANO GRADE: 2015

DATA DA ATIVIDADE	DESCRIÇÃO DA ATIVIDADE	TOTAL DE HORAS
11/04/15	Estudo UML para criação de casos de uso	06:00:00
18/04/15	Estudo UML para criação de casos de uso	04:00:00
18/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	05:00:00
19/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	03:00:00
19/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	06:00:00
25/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	04:00:00
26/04/15	Leitura livro Engenharia de Software, Sommerville (2011)	05:00:00
01/05/15	Estudo dos conceitos globais da Engenharia de Requisitos	04:00:00
02/05/15	Escrita dos conceitos gerais	05:00:00
03/05/15	Escrita dos conceitos gerais	06:00:00
08/05/15	Escrita dos conceitos gerais	04:00:00
09/05/15	Escrita dos conceitos gerais	05:00:00
10/05/15	Escrita dos conceitos gerais	06:00:00
16/05/15	Escrita dos conceitos gerais	03:00:00
17/05/15	Escrita da descrição de atividades	04:00:00

TOTAL DE HORAS ATRIBUÍDAS: 70:00:00