

Universidade de Brasília - UnB
Faculdade UnB Gama - FGA
Requisitos de Software - 201308

Relatório de Projeto

Grupo: 7

Bruno Contessotto Bragança Pinheiro

Eduardo Henrique Fonseca Moreira

Omar Faria dos Santos Junior

Ricardo Lupiano Andrade

Orientador:

George Marsicano Corrêa, MSc.

Brasília, DF
Maio de 2015



Bruno Contessotto Bragança Pinheiro - 09/0107853

Eduardo Henrique Fonseca Moreira - 13/0008371

Omar Faria dos Santos Junior - 13/0015920

Ricardo Lupiano Andrade - 13/0016969

Relatório de Projeto

Trabalho referente ao relatório de projeto
da primeira entrega da materia de Engenharia de Requisitos - 201308 do curso de Engenharia de *Software* da Universidade de Brasília - UnB

Universidade de Brasília - UnB

Faculdade UnB Gama - FGA

Professor Orientador: George Marsicano Corrêa, MSc.

Brasília, DF

Maio de 2015

Índice

1	Cronograma	6
2	Justificativa da Abordagem	6
2.1	Processo Unificado	6
2.2	SAFe	6
2.3	Resultados Obtidos	6
3	Processo de Engenharia de Requisitos	7
4	Técnicas de Elicitação	8
5	Rastreabilidade	8
5.1	Rastreabilidade Vertical	8
5.2	Rastreabilidade Horizontal	9
6	Atributos de Requisitos	9
6.1	Prioridade	10
6.2	Valor	10
6.3	Dificuldade	10
7	Ferramentas de Gerência de Requisitos	10
7.1	Ferramentas Analisadas	10
7.2	Ferramenta Escolhida: codeBeamer	11

Lista de Tabelas

1	Escolha da Metodologia	7
---	----------------------------------	---

Lista de Imagens

1	exemplo de rastreabilidade vertical e horizontal	8
2	exemplo de rastreabilidade vertical de tema de investimento à histórias de usuário.	9
3	exemplo de rastreabilidade horizontal entre histórias de usuário.	9

1 Cronograma

(Espaço reservado para a elaboração do cronograma)

2 Justificativa da Abordagem

Para a definição da abordagem foram estudados o Processo Unificado e o *(Scaled Agile Framework)*.

2.1 Processo Unificado

(Espaço reservado para o Processo Unificado)

2.2 SAFe

(Espaço reservado para o SAFe)

2.3 Resultados Obtidos

De acordo com estudos realizados sobre o Processo Unificado e o *Scaled Agile Framework*, do contexto de negócio e das características dos *stakeholders*, chegamos a algumas questões a serem respondidas. São Elas:

- Integração:
 - O time de desenvolvimento poderá se encontrar com alta frequência?
 - O cliente terá disponibilidade alta para encontros?
- Time:
 - O time mudará durante o desenvolvimento do projeto?
 - O time possui experiência?
 - O time possui forte integração?
- Negócio:
 - A estrutura organizacional da empresa é estável?
 - O cliente demanda formalidades?
 - O sistema é crítico?
 - Os requisitos do projeto mudarão com frequência?
 - O cliente demanda entrega contínua de Software?

A partir das perguntas levantadas, foram respondidas, individualmente por cada membro da equipe de desenvolvimento, as perguntas, e chegou-se a conclusão que:

O time apesar de não possuir tamanha experiência, estão motivados a trabalhar com desenvolvimento ágil, e o farão em reuniões frequentes e semanais, além de possuírem forte integração resultante de projetos passados.

O cliente não demanda formalidades, apesar de necessitar de documentação, e, com a possibilidade de mudança nos requisitos, foi optado por contínua entrega de software e um contato próximo com o cliente.

A partir do resultado obtido, foi gerado a seguinte tabela, e optado a abordagem adaptativa *SAFe* para o desenvolvimento do projeto

Itens	Características	Tradicional	Ágil	Descrição
Interação	Reuniões - equipe de desenvolvimento		x	A equipe de desenvolvimento se reunirá com frequência
	Encontro com cliente		x	A equipe de desenvolvimento manterá contato próximo ao cliente
Time	Mudança de equipe de desenvolvimento		x	Não haverá mudanças na equipe de desenvolvimento
	Experiência da equipe		x	A equipe possui experiência com desenvolvimento ágil de software
	Equipe integrada		x	A equipe se conhece e já trabalhou junta em trabalhos anteriores
Negócio	Requisitos mutáveis		x	Provável evolução do sistema após o fim da primeira etapa de projeto.
	Documentação extensiva para manter o sistema		x	(?)Cliente não requer documentação formal/extensa(?)
	Entregas parciais	x		(?)Não há necessidade de entregas parciais do software (?)
	Projeto não é crítico		x	O projeto em desenvolvimento não é crítico, não exigindo que todo o projeto seja elicitado e bem definido no início de seu desenvolvimento

Tabela 1: Escolha da Metodologia

3 Processo de Engenharia de Requisitos

(Espaço reservado para a elaboração do processo de engenharia de requisitos que será executado em T2.)

4 Técnicas de Elicitação

(Espaço reservado para a elaboração das técnicas de elicitação)

5 Rastreabilidade

A rastreabilidade auxilia a engenharia de requisitos no controle dos requisitos, elementos de modelagem e outros artefatos do processo de software. Por meio dela, é possível obter visualizar de onde surgiu tal requisito, quais são suas dependências e ainda quais deles serão afetados quando houver algum tipo de mudança.

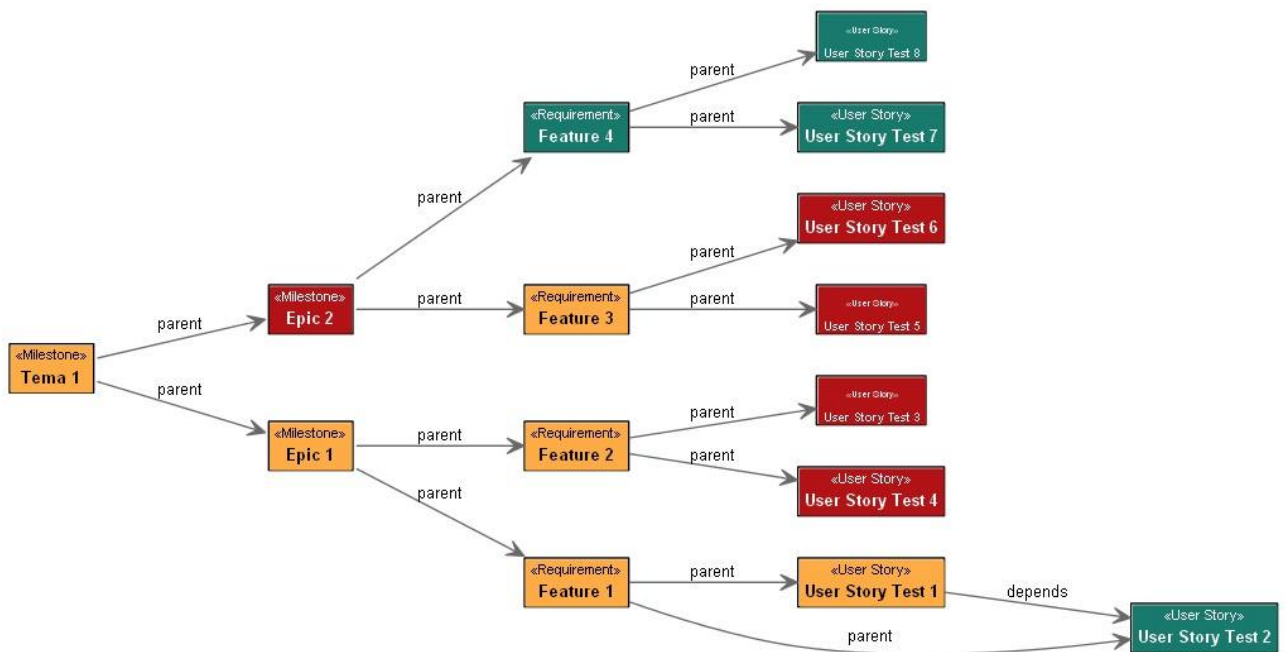


Imagem 1: exemplo de rastreabilidade vertical e horizontal

No desenvolver deste projeto iremos utilizar dois tipos de rastreabilidade, a vertical e a horizontal.

5.1 Rastreabilidade Vertical

A rastreabilidade vertical será utilizada no projeto para identificar a origem dos requisitos, ela está presente nas relações de um nível de abstração e outro.

Dependencies of Tema de Investimento [Export to Office](#)

Tema de Investimento	Epic	Features	User Stories
Req7 → TI-1127 Tema 1 ACTIVE	Req7 → E-1115 Epic 1 IN PROGRESS	Req7 → FEAT-1116 Feature 1 ACTIVE	Req7 → US-1078 User Story Test 1 IN PROGRESS
		Req7 → FEAT-1123 Feature 2 ACTIVE	Req7 → US-1079 User Story Test 2 DONE
	Req7 → E-1126 Epic 2 TODO	Req7 → FEAT-1125 Feature 4 CLOSED	Req7 → US-1117 User Story Test 3 TODO
		Req7 → FEAT-1124 Feature 3 ACTIVE	Req7 → US-1118 User Story Test 4 TODO
			Req7 → US-1121 User Story Test 7 DONE
			Req7 → US-1122 User Story Test 8 DONE
			Req7 → US-1119 User Story Test 5 TODO
			Req7 → US-1120 User Story Test 6 TODO

Imagem 2: exemplo de rastreabilidade vertical de tema de investimento à histórias de usuário.

5.2 Rastreabilidade Horizontal

A rastreabilidade horizontal será utilizada no projeto para identificar as dependencias entre um requisito e outro de um mesmo nível de abstração.

Dependencies of User Stories [Export to Office](#)

User Stories	User Stories
Req7 → US-1122 User Story Test 8 DONE	Req7 → US-1078 User Story Test 1 IN PROGRESS
Req7 → US-1121 User Story Test 7 DONE	Req7 → US-1079 User Story Test 2 DONE
Req7 → US-1120 User Story Test 6 TODO	
Req7 → US-1119 User Story Test 5 TODO	
Req7 → US-1118 User Story Test 4 TODO	
Req7 → US-1117 User Story Test 3 TODO	
Req7 → US-1079 User Story Test 2 DONE	
Req7 → US-1078 User Story Test 1 IN PROGRESS	

Imagem 3: exemplo de rastreabilidade horizontal entre histórias de usuário.

6 Atributos de Requisitos

Atributos de requisitos são propriedades dos mesmos e armazenam informações adicionais. Para o desenvolvimento deste projeto, foram escolhidos os seguintes atributos:

- Prioridade
- Valor
- Dificuldade
- outro..?

6.1 Prioridade

O atributo **prioridade** indica a ordem em que as histórias de usuários deverão ser desenvolvidas. A prioridade da história de usuário pode ser definida em:

- Alta, Média, Baixa ou Indefinida.

6.2 Valor

O atributo **valor** indica o valor agregado de cada história de usuário para os *stakeholders*. O valor de uma história de usuário pode ser definido em:

- Alto, Médio, Baixo ou Indefinido.

6.3 Dificuldade

O atributo **dificuldade** indica o nível de esforço necessário para o desenvolvimento da história de usuário. A dificuldade de uma história de usuário pode ser definido em:

- Alta, Média, Baixa ou Indefinida.

7 Ferramentas de Gerência de Requisitos

7.1 Ferramentas Analisadas

Foram analisadas várias ferramentas de gerência de requisitos, dentre elas, ferramentas web com tempo limite de uso (dentre 7 a 1 mês) como: RequirementOne; SpiraTest; Jira; Visure Requirement; InteGREAT. Ferramentas web free, porém com recursos limitados, como a Rally e o ReqView, e ferramentas onde é necessário a instalação de vários pacotes na máquina local, o que torna a sua instalação difícil ou até mesmo não viável, como a IBM Rational DOORS, Axiom 4, codeBeamer e a enterprise Architect.

Para escolha da ferramenta, foi avaliado se a ferramenta possuía os seguintes quesitos:

- possibilidade de abstração a nível de portfólio;
- possibilidade de abstração a nível de programa;
- possibilidade de abstração a nível de time;
- matriz de Rastreabilidade:
 - rastreabilidade Horizontal;
 - rastreabilidade Vertical;
- tabela de Atributo de Requisitos:

- implementação de novos atributos;
- controle de backlog;

Todas as ferramentas citadas acima que possuíam um tempo limitado de uso menor ou igual a um mês foram descartadas, apesar de tais ferramentas serem as mais fáceis de serem usadas. As ferramentas IBM Rational DOORs e Axiom 4 foram estudadas através de tutoriais e várias tentativas de instalações foram realizadas sem sucesso, tanto no ambiente linux quanto no windows.

As ferramentas ReqView e Enterprise Architect foram implementadas com êxito, porém a reqview é muito simples e não atendia às necessidades do projeto, já a Enterprise Architect é uma ferramenta especializada para projeto com abordagens tradicionais, sendo difícil a utilização para abordagem ágil, com isso ambos também foram descartadas, restando apenas as ferramentas Rally e a codeBeamer.

A ferramenta RALLY foi desenvolvida para auxiliar desenvolvedores na criação de grandes projetos de abordagem ágil, com grande foco na metodologia lean e no SAFe. Sua versão free permite registro e rastreabilidades de abstrações no nível de time (para pequenos projetos), deixando abstrações a nível de programa e portfólio apenas para contribuidores, com isso chegamos a nossa escolha final da ferramenta, a codeBeamer.

7.2 Ferramenta Escolhida: codeBeamer

A ferramenta codeBeamer é uma ferramenta de gerência de requisitos e de projeto gratuita, com limite a um projeto com duração de um ano, e tal projeto pode ser compartilhado entre 25 a 50 pessoas.

Ela permite o usuário montar qual tipo de rastreabilidade ele necessita a partir dos identificadores criados ou pré-existentes (histórias de usuário, features, epicos, temas, entre outros) permitindo assim uma rastreabilidade tanto na horizontal quanto na vertical. O codeBeamer permite ainda a criação de novos atributos, tornando possível a visualização de uma matriz de atributos de requisitos completa a partir da necessidade do usuário.

A ferramenta codeBeamer conseguiu atender a todos os quesitos exigidos para a implementação deste projeto, se tornando assim, a ferramenta escolhida.