



**UNIVERSIDADE FEDERAL DO CEARÁ**  
**CAMPUS DE RUSSAS**  
**CURSO DE ENGENHARIA DE SOFTWARE**

**LUCAS RAFAEL ALVES OLIVEIRA**

**GUIA DE METODOLOGIAS E BOAS PRÁTICAS EM ENGENHARIA  
DE REQUISITOS**

**RUSSAS - CE**  
**2021**

**LUCAS RAFAEL ALVES OLIVEIRA**

**GUIA DE METODOLOGIAS E BOAS PRÁTICAS EM ENGENHARIA  
DE REQUISITOS**

Trabalho de Conclusão de Curso  
apresentado ao Curso de Bacharelado em  
Engenharia de Software do Campus de  
Russas da Universidade Federal do Ceará,  
como requisito parcial para obtenção do  
grau de bacharel em Engenharia de  
Software.

Orientador: Prof. Ms. José Osvaldo  
Mesquita Chaves.

**RUSSAS – CE**

**2021**

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Biblioteca Universitária

Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

- A48g Alves Oliveira, Lucas Rafael.  
Guia de Metodologia e Boas Práticas em Engenharia de Requisitos / Lucas Rafael Alves Oliveira. –  
2021.  
180 f. : il. color.
- Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Russas,  
Curso de Engenharia de Software, Russas, 2021.  
Orientação: Prof. Me. José Osvaldo Mesquita Chaves.
1. Guia Técnico. 2. Engenharia de Requisitos. 3. Qualidade de Requisitos. 4. Processos de Engenharia  
de Requisitos. I. Título.

CDD 005.1

---

**LUCAS RAFAEL ALVES OLIVEIRA**

**GUIA DE METODOLOGIAS E BOAS PRÁTICAS EM ENGENHARIA  
DE REQUISITOS**

Trabalho de Conclusão de Curso  
apresentado ao Curso de Bacharelado em  
Engenharia de Software do Campus de  
Russas da Universidade Federal do Ceará,  
como requisito parcial para obtenção do  
grau de bacharel em Engenharia de  
Software.

Orientador: Prof. Ms. José Osvaldo  
Mesquita Chaves.

Aprovado em: 02/04/2021.

**BANCA EXAMINADORA**

---

Ms. José Osvaldo Mesquita Chaves (Orientador)  
Universidade do Estado do Rio Grande do Norte (UERN)

---

Dra. Patrícia Freitas Campos de Vasconcelos  
Universidade de Fortaleza (UNIFOR)

---

Dr. Jacilane de Holanda Rabelo  
Universidade Federal do Amazonas (UFAM)

## RESUMO

Os requisitos são a base para o software. Um requisito que não seja bem entendido, bem escrito e bem gerenciado pode trazer grandes impactos ao projeto. Isso quando a empresa leva em consideração em seus processos atividades de engenharia de requisitos. Algumas empresas que têm bastante dificuldade em estabelecer um processo de qualidade na área de engenharia de requisitos e muitas vezes ficam perdidas e sem saber qual melhor método utilizar, acabam começando a fazer projetos sem uso de um processo de qualidade, partindo diretamente para programação e tornando isso um hábito na empresa deixando o produto com a qualidade bem abaixo do que os seus clientes desejavam. Assim, neste trabalho é proposto um guia que selecione um processo de engenharia de requisitos, com base na quantidade de membros nas equipes, na experiência dos membros das equipes em relação as áreas de engenharia de requisitos, se é uma empresa de TI, ou com setor de TI, ou é Startup, ou um grupo de faculdade. Além de levar em consideração o tempo de desenvolvimento que os projetos demandam, ou se é focada em fazer software por encomenda ou software como serviço.

**Palavras-chaves:** Guia Técnico; Engenharia de Requisitos; Qualidade de Requisitos; Processos de Engenharia de Requisitos.

## ABSTRACT

The requirements are a basis for the software. A requirement that is not well understood, well written and well managed can have major impacts on the project. This is when the company takes requirements engineering activities into account in its processes. Some companies that have a hard time establishing a quality process in the requirements engineering area and are often lost and not knowing which method to use best, end up starting to do projects without using a quality process, starting directly for programming and processing this is a habit in the company leaving the product with the quality well below what its customers wanted. Thus, this work proposes a guide that selects a requirements engineering process, based on the number of members in the teams, the experience of the team members in relation to the requirements engineering areas, whether it is an IT company, or with IT sector, or is Startup, or a group of faculty. In addition to taking into account the development time that projects require, or whether it is focused on making software to order or software as a service.

**Keywords:** Technical Guide; Requirements Engineering; Requirements Quality; Requirements Engineering Processes.

## LISTA DE IMAGENS

Figura 1 - O Processo de Engenharia de Requisitos.....	12
Figura 2 - Os cinco níveis de maturidade do CMMI.....	13
Figura 3 - Evolução dos processos nos níveis de maturidade.....	15
Figura 4 - Processo de desenvolvimento de um Guia de metodologias e boas práticas em engenharia de requisitos.....	18
Figura 5 - Planilha dinâmica com os perfis de empresas e os processos desenvolvidos para uso do guia.....	22

## **LISTA DE TABELAS**

Tabela 1 - Apresentação de ações que cada trabalho realiza.....	17
---	----

## SUMÁRIO

1	INTRODUÇÃO .....	8
2	OBJETIVOS .....	10
2.1	Objetivo geral.....	10
2.2	Objetivos específicos .....	10
3	FUNDAMENTAÇÃO TEÓRICA.....	11
3.1	Processo .....	11
3.2	Engenharia de Requisitos.....	12
3.3	CMMI.....	13
3.4	MPS.Br.....	14
4	TRABALHOS RELACIONADOS .....	16
5	PROCEDIMENTOS METODOLÓGICOS.....	18
6	ELABORAÇÃO DOS PERFIS E DO GUIA .....	21
7	RESULTADOS OBTIDOS .....	25
8	CONCLUSÃO E TRABALHOS FUTUROS .....	28
	REFERÊNCIAS .....	29
	APÊNDICES.....	

## 1 INTRODUÇÃO

Os requisitos são a base para qualquer projeto de software, pois eles representam as necessidades das partes interessadas. Para Pressman (2011, c. 5) desenvolver um sistema elegante que resolva o problema de forma errada não atende as necessidades de ninguém, logo, o sistema a ser desenvolvido deve suprir essas necessidades. Portanto os requisitos são de suma importância para o desenvolvimento do projeto, pois a partir deles pode-se realizar o planejamento do projeto, o gerenciamento dos riscos, a realização de testes, o controle de mudanças, entre outras ferramentas fundamentais para o sucesso do projeto.

Porém, são muitos os casos de projetos de software cancelados ou que fracassaram por não atenderem completamente as necessidades dos clientes e excederem o prazo e o orçamento estimados (SIMÕES, 2017). Portanto, saber como lidar com as necessidades dos clientes se mostra uma tarefa difícil, além de que manter e gerenciar os requisitos coletados pode se tornar uma tarefa complexa e confusa, de modo que saber como lidar com essas tarefas e escolher como realizá-las não é uma tarefa trivial.

Em (LeandroMTR, 2016) em um artigo intitulado “CMMI usar ou não usar?” enfatiza que o CMMI deve ser considerado, mesmo que não adotado, pois traz uma discussão abrangente sobre as atividades que devem estar no desenvolvimento de um software e que abraçar seus conceitos traz uma maior maturidade para os processos. Já Silva (2013) fala sobre a construção do MPS.Br como um modelo de qualidade para melhorar a capacidade de desenvolvimento de empresas brasileiras. Portanto, utilizar tais modelos reconhecidos vai proporcionar a criação de novos modelos com níveis de qualidade maiores.

Mas o que é engenharia de requisitos e em qual parte do projeto ela está? Para (Higor, 2013) em seu artigo, a Engenharia de Requisitos é o processo pelo qual os requisitos de um software são coletados, analisados, documentados e gerenciados ao longo de todo o ciclo de vida do software. Com isso é possível perceber que os requisitos possuem um papel fundamental para o desenvolvimento de software. Sendo eles uma das principais medidas de sucesso de um software, visto que se o sistema desenvolvido atende aos seus requisitos logo o software estará de acordo com as necessidades do cliente. Porém, como saber quais práticas, processos e atividades melhor se adequam ao projeto? Uma vez que é difícil coletar essas informações se o profissional não tiver experiência com essa tarefa, pois muitas vezes o conteúdo necessário é de difícil acesso ou está espalhado e fragmentado em vários lugares.

Então, pensando na necessidade de ter acesso a esse conteúdo de forma mais eficiente é que esse trabalho foi pensado, para ser um guia, onde seja possível encontrar boas práticas da engenharia de requisitos, processos ou mesclagem de processos que melhor se adequem ao projeto. Visando melhorar o desempenho de pequenas empresas e de profissionais que ainda não tenham experiência no mercado, além de oferecer exemplos de documentos e artefatos que o processo pode gerar.

O trabalho usará os modelos ágeis como base, pois eles são amplamente utilizados, segundo (Borges, 2017), em uma matéria no site IGTI BOLG, no ano de 2016, 55% das empresas do Brasil adotaram metodologias ágeis, e levando em consideração o passar dos anos esse número vem crescendo gradativamente.

Diante do contexto apresentado, o presente trabalho propõe a criação de um guia que poderá ser utilizado por professores, alunos, profissionais e empresas de TI. Que tem como objetivo sugerir métodos, processos, artefatos e boas práticas no âmbito da engenharia de requisitos. Com o uso do guia será possível encontrar recomendações de acordo com as características das empresas ou dos projetos que estão em desenvolvimento.

A organização deste trabalho encontra-se da seguinte forma: o Capítulo 2 apresenta os objetivos (gerais e específicos). No Capítulo 3 é apresentada a fundamentação teórica, que tem como propósito ajudar a compreender termos essenciais tratados durante o trabalho. O Capítulo 4 trata de discutir sobre os trabalhos relacionados. O Capítulo 5 define o procedimento metodológico, explicando cada etapa da pesquisa científica, além de apresentar o cronograma para realização do trabalho. O Capítulo 6, mostra, a criação do guia, desde os perfis encontrados, processos desenvolvidos e filtros usados. O Capítulo 7 mostra os resultados obtidos da aplicação do guia no N2S (Núcleo de Soluções em Software) na UFC Campus de Russas, e na Empresa Recicla Go. O Capítulo 8, mostra as conclusões deste trabalho, além de conter os trabalhos futuros.

## **2 OBJETIVOS**

### **2.1 Objetivo geral**

O presente trabalho tem como objetivo principal fornecer um guia de metodologias e boas práticas em engenharia de requisitos, que auxilie profissionais, professores e alunos.

### **2.2 Objetivos específicos**

- Identificar os vários perfis de empresas e acadêmicos.
- Apoiar na escolha de métodos de engenharia de requisitos de acordo com o perfil de cada profissional.
- Apoiar o processo de elicitação de requisitos.
- Usar os modelos de qualidade para garantir a qualidade do processo.

### 3 FUNDAMENTAÇÃO TEÓRICA

Este trabalho trata conceitos de engenharia de requisitos e suas metodologias, bem como os processos abordados, boas práticas e qualidade de requisitos. Deste modo, este capítulo apresenta os principais conceitos que fundamentam essa pesquisa.

#### 3.1 Processo

Para Sommerville (2011, c. 2, p. 18):

Um processo de software é um conjunto de atividades relacionadas que levam à produção de um produto de software. Essas atividades podem envolver o desenvolvimento de software a partir do zero em uma linguagem padrão de programação como Java ou C.

Ainda para Sommerville (2011, c.2) um processo possui vários elementos que devem ser discutidos quando se fala de processo em Engenharia de Software, entre eles, as atividades que se concentra em: especificação de software, onde suas funcionalidades, restrições e funcionamento são previamente definidas, o projeto e implementação do software, onde o software deve ser produzido para atender as especificações, definidas na atividade anterior, a validação do software, onde, o software deve ser validado para garantir que ele atende as especificações e necessidades do cliente e a evolução do software, onde o software evoluir para atender às necessidades de mudanças do cliente, que podem surgir.

Conforme Pressman (2011, c. 2, p.52) “Processo é um conjunto de atividades, ações e tarefas realizadas na criação de algum produto de trabalho”.

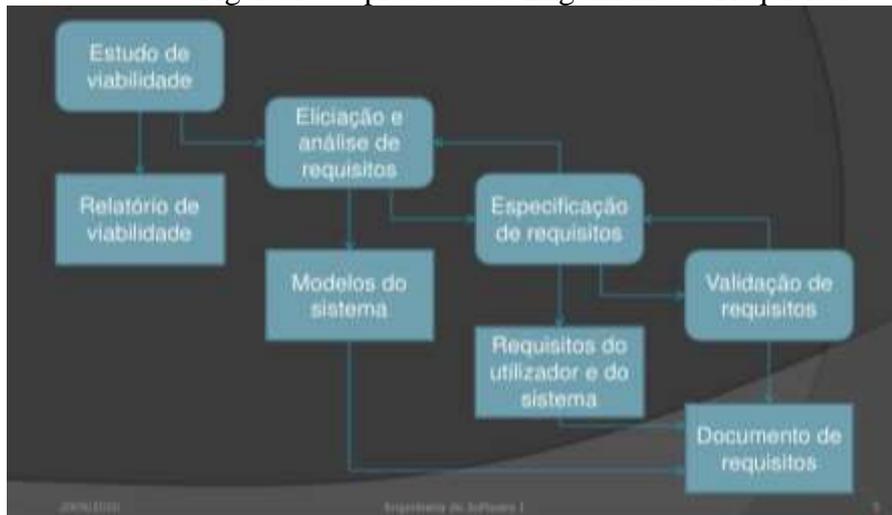
Além das atividades há também o produto, que são os resultados obtidos através da realização das atividades do processo, os papéis, que são as responsabilidades das pessoas envolvidas no processo e a pré e pós condições, que são declarações verdadeiras antes e depois de uma atividade do processo ou da construção de um produto.

Logo, neste trabalho, iremos estudar as atividades que dizem respeito a engenharia de requisitos de alguns processos, ou seja, será focado nas atividades de especificação de software, especialmente processos ágeis, tendo em vista a qualidade dos requisitos e as características das empresas e profissionais que usarem o guia.

### 3.2 Engenharia de Requisitos

Como podemos ver na Figura 1, a Engenharia de Requisitos é um processo que representa toda a parte de planejamento do software e sua viabilidade, dando uma visão sobre o software antes de seu desenvolvimento.

Figura 1 - O processo da Engenharia de Requisitos



Fonte: Slideshare, Eng.<sup>a</sup> do Software - 3. Processos da engenharia de requisitos 2010

De acordo com Pressman (2011, c. 5, p. 127) “O amplo espectro de tarefas e técnicas que levam a um entendimento dos requisitos é denominado engenharia de requisitos”.

Para Vazquez (2016) a engenharia de requisitos consiste no uso sistemático e repetitivo de atividades de obtenção, documentação e manutenção de requisitos de software que atendam aos objetivos do negócio e que tenham qualidade.

Portanto, devemos entender o que são metodologias ágeis e como funciona a engenharia de requisitos com elas. André (2011) sugere que as metodologias ágeis focam mais em pessoas e em interações, aumentando o contato entre os membros da equipe e o cliente permitindo uma melhor comunicação, não foca muito em documentos extensos, porém não deixa de lado, só os deixa bem mais simplificados e permite que a equipe responsável pelo projeto possa reagir a mudança bem mais rápido e com mais eficácia, logo é uma metodologia que permite ter uma visão clara do resultado do software antes dele estar pronto.

Já Igor (2014), em um artigo com o título, Engenharia de Requisitos Ágeis, fala sobre a engenharia de requisitos, nas técnicas de métodos ágeis se constitui de técnicas relacionadas a evitar problemas como especificação de requisitos incompleta, baixo envolvimento do cliente, expectativas não realizadas, entre outros.

Segundo Sommerville (2011, c. 4, p. 57) “Os requisitos de um sistema são as descrições do que o sistema deve fazer, os serviços que oferecem e as restrições a seu funcionamento”. Logo podemos entender os requisitos como necessidade, exigência, desejo ou solicitação, e pode considerar que ele é o começo de “tudo” e a base para o “todo” do sistema, portanto o seu entendimento é vital para um sistema de qualidade.

### 3.3 CMMI

O Modelo de Capacidade e Maturidade Integrado (CMMI - *Capability Maturity Model Integration*) é um conjunto de práticas que orienta a implementação de atividades com o objetivo de uma meta preestabelecida, aumentando a maturidade organizacional de uma empresa. Contando com 5 níveis como mostra a Figura 2.

Figura 2 - Os cinco níveis de maturidade do CMMI



Fonte: ISD Brasil.

Existindo, ainda, mais um que é quando a empresa não tem nenhum critério, sendo assim 6 níveis:

- **Nível 0 - Incompleto:** o nível maturidade 0 chamado de incompleto, nele o trabalho é feito de forma aleatória, onde não é seguido um padrão ou método na construção dos projetos e com resultados incompletos.

- Nível 1 - Inicial: o nível de maturidade 1 chamado de inicial, apesar dos trabalhos serem completados na maioria das vezes são entregues atrasados ou com estouro no orçamento.
- Nível 2 - Gerenciado: o nível de maturidade 2 chamado de gerenciado, neste nível os projetos são planejados, realizados, medidos e controlados.
- Nível 3 - Definido: o nível de maturidade 3 chamado de definido, um nível mais proativo que os outros, neste nível são definidos padrões de processos que servirão de orientação para a organização, nos projetos, programas e portfólios.
- Nível 4 - Gerenciado Quantitativamente: o nível de maturidade 4 chamado de gerenciado quantitativamente, organizações e projetos de alta maturidade que usam análise quantitativa e estatística para determinar, identificar e gerenciar a tendência de dispersão central, para entender e abordar a estabilidade de processos e como impactam nos objetivos de obtenção de qualidade.
- Nível 5 - Otimização: o nível de maturidade 5 chamado de otimização, organizações no nível 5 buscam a melhoria dos processos, para torná-los flexíveis capazes de responder a oportunidades e mudanças.

Com base nos conhecimentos através do estudo do CMMI, será usado no trabalho apenas o conhecimento relacionado a Engenharia de Requisitos, como atividades, artefatos, entre outros para a elaboração de processos baseados na maturidade proposta.

### **3.4 MPS.Br**

É o modelo de qualidade brasileiro, que se adequa aos mais diferentes perfis de empresas com suas diferentes características, com foco em micro, pequenas e médias empresas. Os modelos se adequam aos padrões de qualidade aceitos internacionalmente e que têm como pressuposto o aproveitamento de boas práticas representadas nos padrões de modelos de melhorias de processos já disponíveis.

O modelo se baseia principalmente nas normas ISO, e possui vários níveis de maturidade indo do G ao A como se pode ver na Figura 3, porém no trabalho só será abordada a parte que diz respeito a engenharia de requisitos que se estende pelos níveis, para a elaboração dos processos propostos no guia, assim os processos seguirão as métricas de qualidade propostas pelo MPS.Br.

Figura 3 - Evolução dos processos nos níveis de maturidade



Fonte: MPS.BR-Guia Geral MPS de Software 2020

#### 4 TRABALHOS RELACIONADOS

Este trabalho teve como ponto de partida o trabalho de conclusão de curso da aluna ALENCAR (2019) que desenvolveu um guia para selecionar métodos de avaliação para tecnologias de softwares *M-learning*. Por sua vez, o presente trabalho visa reunir em um só lugar processos de engenharia de requisitos, permitindo aos utilizadores do guia obterem mais qualidade na área e, conseqüentemente, mais sucesso nos seus projetos.

POLCELLI (2005) fez um estudo sobre a relevância de boas práticas em engenharia de requisitos, mostrando que na construção de um software os requisitos podem ter um impacto negativo no final da construção, mostrando, também, que empresas podem minimizar esses impactos usando boas práticas definidas em modelos de qualidade.

Por sua vez, o trabalho de CINTRA (2006) implementou um processo de engenharia de requisitos baseado no Processo Unificado da *Rational* (RUP) que alcança o nível 3 de maturidade do CMMI e inclui a utilização de práticas ágeis. A partir do trabalho citado, será possível, para o guia, a elaboração dos processos sugeridos que mais se adequem as empresas seguindo padrões de qualidade amplamente aceitos, tanto pelo CMMI, como pelo MPS.Br.

O trabalho desenvolvido por JAQUEIRA (2013) propôs um modelo que usa um mapeamento de estórias de usuários em modelos gráficos, para melhorar o processo de engenharia de requisitos em modelos ágeis. Esse modelo servirá de base para processos de engenharia de requisitos em que a velocidade no desenvolvimento seja necessária e a equipe seja reduzida.

Em MARTINS (2001) é proposta uma metodologia de elicitação de requisitos, onde o método é centrado nas atividades realizadas pelos usuários do sistema, para que possa ser feita a coleta de requisitos de forma mais eficaz, acompanhando e compreendendo melhor os requisitos. Onde o modelo proposto se divide em três etapas principais, sendo elas: Divisão de problemas em atividades, Delimitação do contexto das atividades e a Descrição da estrutura hierárquica das atividades, onde cada etapa vai se tornando cada vez mais específica. Este trabalho traz uma visão de metodologias que podem ser usadas, além das formas convencionais como entrevista, questionário, observação, entre outras, assim contribuindo com a construção de novos processos.

CAVALCANTE (2018) propôs um modelo baseado na combinação de metodologias ágeis. Utilizando essa abordagem é possível usar o modelo proposto como base para a construção de processos que seguem os modelos ágeis, incrementando os modelos de maturidade propostos pelo CMMI e MPS.Br, mas sem perder a base dos modelos ágeis.

Porém, levando em consideração somente a parte do modelo que diz respeito a engenharia de requisitos, que será usado em conjunto aos modelos e padrões de qualidade proposto pelo CMMI ou MPS.Br e assim utilizar como proposta de modelos no guia.

A Tabela 1 exibe o que cada trabalho relacionado contempla e, assim, traça um paralelo com o trabalho proposto nesta pesquisa. É importante destacar que os estudos citados contribuíram de forma significativa ao desenvolvimento desta proposta de pesquisa, dando a inspiração para a construção do guia, bem como, servindo de base, para o conteúdo que será abordado no guia.

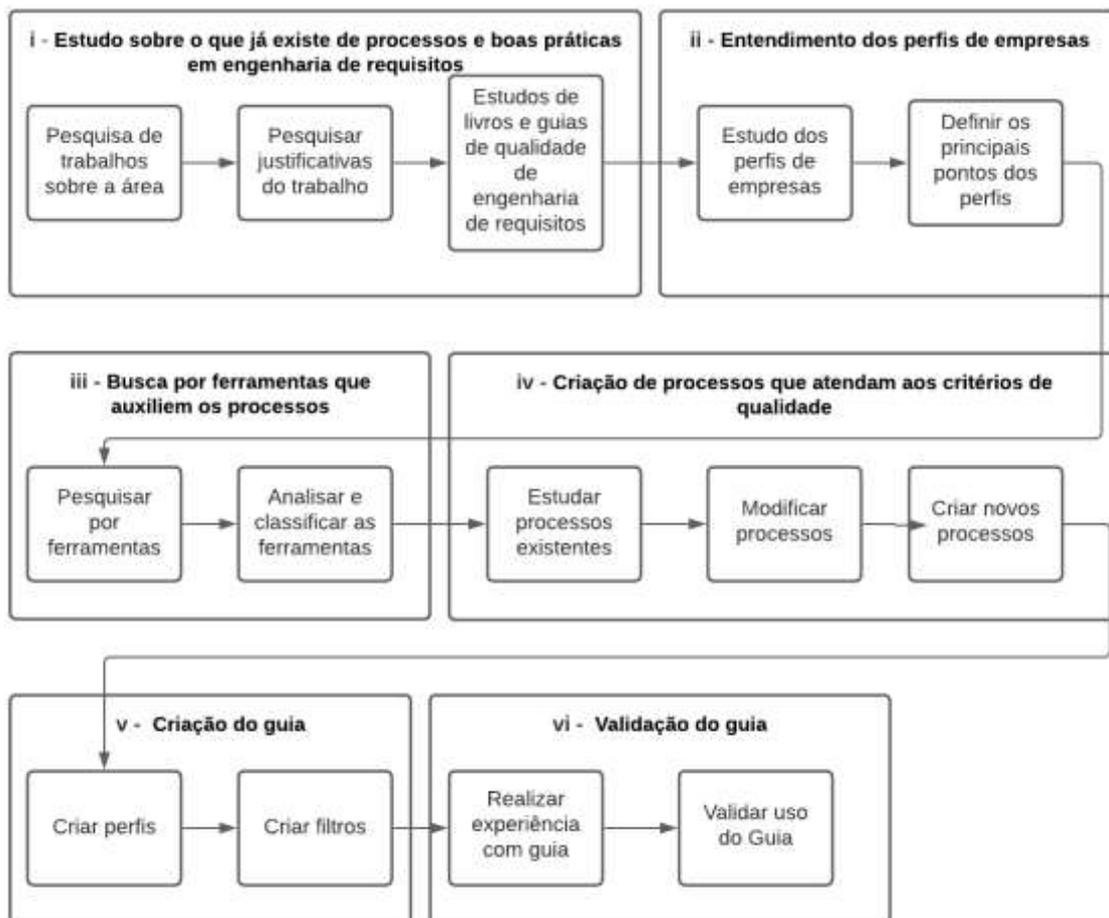
Tabela 1 - Apresentação de ações que cada trabalho realiza.

	É um guia prático, propondo um uso mais simples e otimizado.	Capaz de ser dinâmico.	Propõe soluções para perfis de usuários diferentes.	Propõem processos ou modelo.	Seguem o MPS.Br	Uso de boas práticas em Engenharia de Requisitos.	Seguem o CMMI
ALENCAR (2019)	X	X	X				
POLCELLI (2005)						X	
CINTRA (2006)				X		X	X
JAQUEIRA (2013)				X		X	
MARTINS (2001)				X		X	
CAVALCANTE (2018)				X			
Este trabalho	X	X	X	X	X	X	X

## 5 PROCEDIMENTOS METODOLÓGICOS

Este capítulo descreve as etapas que foram seguidas para a realização desta pesquisa. Que se deu através de diversos estudos e pesquisas, durante elas foram identificados processos e perfis das empresas, além de propor mais outros processos baseados em critérios de qualidade abordados pelo CMMI e MPS.Br. Toda esta informação será inserida e organizada em planilhas permitindo a criação do guia através de uma tabela dinâmica e diversos filtros. Conforme pode ser visto na Figura 4, o processo de desenvolvimento do guia é composto pelas seguintes macro atividades:

Figura 4 - Processo de desenvolvimento de um Guia de metodologias e boas práticas em Engenharia de Requisitos



Fonte: elaborado pelo autor.

Cada macro atividade possui um conjunto de tarefas a serem realizadas, conforme explicado a seguir:

**i) Estudo sobre o que já existe de processos e boas práticas em engenharia de requisitos:** esta macro atividade tem como objetivo o estudo de trabalhos que já existem e

que podem agregar valor à pesquisa, bem como embasar a justificativa do trabalho e o estudo do que já é comprovado de qualidade em engenharia de requisitos. Ela é composta pelas seguintes tarefas:

- **Pesquisa de trabalhos sobre a área:** tem como objetivo extrair os trabalhos que possam agregar valor e servir de base para o guia, fazendo pesquisas em bancos de dados e repositórios de trabalhos acadêmicos.
- **Pesquisar justificativas do trabalho:** tem como objetivo mostrar o contexto real da área de engenharia de requisitos que se encontra, mostrando que a engenharia de requisitos é importante e que tem grandes impactos, através de estudos de artigos e pesquisas.
- **Estudos de livros e guias de qualidade em engenharia de requisitos:** o objetivo do estudo é ter a maior compreensão possível sobre critérios de qualidade e processos que atendam a esses critérios.

**ii) Entendimento dos perfis de empresas:** esta macro atividade tem como objetivo o estudo dos perfis que são importantes para a construção do guia. As seguintes tarefas compõem esta atividade:

- **Estudo dos perfis das empresas:** nesta tarefa será feito o levantamento dos perfis dos usuários do guia, através de pesquisas, entrevista e conversas, como o próprio N2S e uma nova startup Recicla Go que trabalha com um aplicativo de reciclagem, além de estudos de depoimentos de pessoas que trabalham em empresas.
- **Definir os principais pontos dos perfis:** nesta tarefa os dados coletados são separados e selecionados de acordo com características que serão importantes para o guia e seu filtro.

**iii) Busca por ferramentas que auxiliem os processos:** esta macro atividade tem como objetivo a busca por ferramentas que possam ajudar nos processos desenvolvidos para o guia, para assim ajudar a manter os níveis de qualidade de cada processo. Possuindo as seguintes tarefas:

- **Pesquisar por ferramentas:** nesta tarefa será realizada uma pesquisa por ferramentas existentes que possam ajudar no gerenciamento do processo de engenharia de requisitos.
- **Analisar e classificar as ferramentas:** nesta tarefa, cada ferramenta selecionada durante a pesquisa é analisada e classificada, de acordo com os

benefícios que ela pode trazer ao processo, descartando as que se mostrarem complicadas demais e que vão demandar muito esforço e tempo para o aprendizado.

**iv) Criação de processos que atendam aos critérios de qualidade:** esta macro atividade tem como objetivo criar os processos que serão sugeridos no guia e que contenham o máximo de critérios de qualidade do CMMI e MPS.Br.

- **Estudar processos existentes:** esta tarefa tem como objetivo estudar processos já desenvolvidos e testados para verificar a necessidade de modificá-los ou criar novos, dependendo da necessidade.
- **Modificar processos:** nesta tarefa os processos encontrados serão modificados de acordo com as necessidades dos perfis encontrados, os processos serão modificados e adaptados com base no CMMI e MPS.Br.
- **Criar novos processos:** nesta tarefa serão criados processos para os perfis que não foram contemplados pelos processos encontrados e que atendam aos critérios de qualidade do CMMI e MPS.Br.

**v) Criação do guia:** esta atividade compreende todo o processo de “montagem” do guia, usando o Excel, para elaborar uma planilha com as informações coletadas. Compreende as seguintes tarefas:

- **Criar perfis:** onde os perfis que foram encontrados são criados e refinados para serem usados nos filtros, serão feitos perfis com base nos dados coletados dos usuários do guia.
- **Criar filtros:** onde os filtros que vão ligar os perfis aos processos e boas práticas serão definidos e implementados em uma planilha.

**vi) Validação do guia:** esta macro atividade tem como objetivo investigar e afirmar que o guia cumpre com sua finalidade. Para isso serão realizadas as seguintes tarefas:

- **Realizar experiências com o guia:** nesta tarefa os usuários responsáveis por trabalharem com requisitos, utilizarão o guia para a seleção de um método adequado, e com base na experiência, fornecerão um *feedback* sobre o uso.
- **Validar uso do guia:** com base no *feedback* e nos resultados obtidos nos trabalhos, esta tarefa consiste em verificar se o guia cumpre com o que se propõe, através da avaliação e se o processo proposto pelo guia incrementou nos resultados obtidos pelos usuários.

## 6 ELABORAÇÃO DOS PERFIS E DO GUIA

Com base nas entrevistas realizadas com membros de startups, da Include Jr. e da vivência do processo no N2S (Núcleo de Soluções em Software), além de pesquisas de processos já existentes, mostrados no Capítulo 4, a criação do guia de metodologias e boas práticas em engenharia de requisitos, com os perfis de usuários (empresas, professores ou estudantes) foi possível. A seguir a lista das características encontradas.

- **Equipes com membros de:** 1 à 2, 3 à 5 ou 6+ membros;
- **Equipes com divisão de cargos:** Não, todos são desenvolvedores; Sim, porém apenas entre desenvolvedor e analista; Sim, com gerente de projeto, analista de sistema, desenvolvedores e analista de teste; Não, mas a equipe divide as responsabilidades; Sim, a equipe tem um especialista na área.
- **O tipo da empresa:** Empresa com setor de TI; Startup de TI; Grupo da faculdade; Empresa focada em software.
- **Tempo de construção:** Necessidade de ser rápido (em mês); Tempo intermediário (não pode passar de 6 meses); Tempo maior (pode passar de 6 meses, mas nada mais que 1 ano); Projetos grandes (acima de 6 meses, 1 ano).
- **Experiência com a área de requisitos:** Membros sem experiência; há pelo menos um membro que entende do assunto; Todos os membros já tiveram experiências com a área.
- **Especificação da empresa:** A empresa trabalha com software por encomenda, onde é necessário a encomenda de um cliente para dar início ao desenvolvimento; ou a empresa trabalha com software como serviço, onde ela produz por conta própria o software.

Todas essas características foram agrupadas em uma planilha dinâmica, onde o usuário irá selecionar as características de acordo com o perfil da equipe, empresa, grupo ou núcleo que está trabalhando. A Figura 5, mostra como está à disposição da planilha.

Figura 5 – Planilha dinâmica, com os perfis de empresas e os processos desenvolvidos para uso do guia.

Membros	Divisão de cargos	Tipo de empresa	Tempo de construção	Experiencia com a área de requisitos	Especificação
01 à 02	Não, mas a equipe divide as responsabilidades	Empresa com setor de TI	Necessidade de ser rápido( em um mês)	Há pelo menos um membro que entende assunto	A empresa trabalha com software como serviço
03 à 05	Não, todos são desenvolvedores	Empresa focada em software	Projetos grandes( acima de 6 meses)	Membros sem experiência	A empresa trabalha com software por encomenda
6+	Sim, a equipe tem um especialista na área	Grupo da Faculdade	Tempo mais maleável (Não pode passar de 6...)	Todos os membros já tiveram alguma experiênci...	
	Sim, com gerente de projeto, analista de sistema, desen...	Startup de TI	Tempo maleável mais de seis meses, mas nad...		
	Sim, porém apenas entre desenvolvedor e analista				
Resultado					
Modelo de Engenharia de Requisitos Ágil: htt...					
Modelo de Estória de Usuários com Cliente: ht...					
Modelo de Estória de Usuários sem Clientes: h...					
Modelo de Processo Ágil com Cliente: https://...					
Modelo de Processo Ágil sem Cliente: https://...					
Modelo de Processo Ágil Simplificado com Cl...					
Modelo de Processo Ágil Simplificado sem Cl...					

Fonte: elaborado pelo autor.

O Guia foi feito através de uma tabela, e com filtros dinâmicos, para que os usuários possam usar e obter um resultado com base nas características. Todos os processos desenvolvidos se encontram na Seção e Apêndices, ao todo foram elaborados 7 processos, todos baseados no MPS Br e no CMMI.

A seguir o guia foi disponibilizado no Google Drive, porém sempre que for usado ele deve ser baixado e usado no Excel, caso contrário os filtros elaborados não utilizáveis. Link: [https://docs.google.com/spreadsheets/d/1Ed\\_AwHkziL3CQEV4cSLvftpT9AzthDeg/edit?usp=sharing&ouid=111222583058299498978&rtpof=true&sd=true](https://docs.google.com/spreadsheets/d/1Ed_AwHkziL3CQEV4cSLvftpT9AzthDeg/edit?usp=sharing&ouid=111222583058299498978&rtpof=true&sd=true)

Os processos que foram desenvolvidos neste trabalho, foram pensados com base nas características encontrados nas pesquisas e na união dessas características, assim a seleção de características leva a um resultado, que será mostrado a seguir.

- **Modelo de Estória de Usuário com Cliente (Apêndice 1)**, é um modelo baseado em Estórias de Usuários, onde os responsáveis coletam os requisitos dos clientes e as convertem em Estórias, a partir das Estórias a equipe deve elaborar os diagramas e sua descrição, proporcionando uma quantidade menor de documentação, além de ser bem mais reduzido, ideal para equipes pequenas, com 1 ou 2 membros, porém pouco recomendado para usuários sem experiência na área de engenharia de requisitos, pois podem facilmente cometer erros na hora de criar os diagramas a partir das Estórias, o processo levou em consideração o nível mais básico do MPS .Br e o trabalho feito por JAQUEIRA (2013).
- **Modelo de Estória de Usuário com Cliente Implícito (Apêndice 2)**, é um modelo baseado em Estórias de Usuários, onde os responsáveis coletam os

requisitos, porém não possuem clientes específicos, então os requisitos são coletados através de pesquisas de mercado, de entrevistas e questionários com possíveis usuários e de aplicações semelhantes, e após essa coleta os requisitos são convertidos em Estórias, a partir das Estórias a equipe deve elaborar os diagramas e sua descrição, proporcionando uma quantidade menor de documentação, além de ser bem mais reduzido, ideal para equipes pequenas, com 1 ou 2 membros, porém pouco recomendado para usuários sem experiência na área de engenharia de requisitos, pois podem facilmente cometer erros na hora de criar os diagramas a partir das Estórias, o processo levou em consideração o nível mais básico do MPS .Br e o trabalho feito por JAQUEIRA (2013).

- **Modelo de Processo Ágil com Cliente (Apêndice 3)**, é um modelo baseado em uma mesclagem de processos ágeis, como Scrum, XP, entre outros, construído com base nos trabalhos de CAVALCANTE (2018) e CINTRA (2006), com adaptações de boa parte dos níveis G e F do MPS .Br e influenciado pelo conhecimento do CMMI. O processo também teve bastante influência da observação e vivência no N2S, foi desenvolvido principalmente para equipes de pequeno e médio porte, com possível frequência de mudança de integrantes na equipe, é um processo que pode ser usado tanto por usuário com ou sem experiência na área, voltado para empresas que trabalham com clientes, fazendo um trabalho mais tradicional dos modelos ágeis que são usados, porém devido a frequência de mudanças no pessoal, é um processo pensando com foco na agilização da documentação e no seu rastreamento, mantendo sempre na versão mais atualizada.
- **Modelo de Processo Ágil com Cliente Implícito (Apêndice 4)**, é um modelo baseado em uma mesclagem de processos ágeis, como Scrum, XP, entre outros, construído com base nos trabalhos de CAVALCANTE (2018) e CINTRA (2006), com adaptações de boa parte dos níveis G e F do MPS .Br e influenciado pelo conhecimento do CMMI. O processo também teve bastante influência da observação e vivência no N2S, foi desenvolvido principalmente para equipes de pequeno e médio porte, com possível frequência de mudança de integrantes na equipe, é um processo que pode ser usado tanto por usuário com ou sem experiência na área, voltado para empresas que não trabalham com clientes, focando assim em pesquisas, entrevista e questionários com possíveis usuário,

além de ter toda a avaliação de outras soluções, porém devido a frequência de mudanças no pessoal, é um processo pensando com foco na agilização da documentação e no seu rastreamento, mantendo sempre na versão mais atualizada.

- **Modelo de Processo Ágil Simplificado com Cliente (Apêndice 5)**, é um processo voltado para empresas que tenham menos frequência nas mudanças de equipes e com mais experiência, mas ainda assim trabalha com documentação ampla, mas focando em manter a construção dos documentos de forma ágil, mas em menor volume que os processos anteriores, também baseado no trabalho de CAVALCANTE (2018) e nos níveis G e F do MPS .Br. O processo foca principalmente em como manter a documentação simples e em como rastrear as informações, além de ser voltado para empresas que trabalham com clientes.
- **Modelo de Processo Ágil Simplificado com Cliente Implícito (Apêndice 6)**, é um processo voltado para empresas que tenham menos frequência nas mudanças de equipes e com mais experiência, mas ainda assim trabalha com documentação ampla, mas focando em manter a construção dos documentos de forma ágil, mas em menor volume que os processos anteriores, também baseado no trabalho de CAVALCANTE (2018) e nos níveis G e F do MPS .Br. O processo foca principalmente em como manter a documentação simples e em como rastrear as informações, além de ser voltado para empresas que não trabalham com clientes, voltando a parte de coleta para uma análise de mercado e de aplicações semelhantes.
- **Modelo de Engenharia de Requisitos Ágil (Apêndice 7)**, é um processo voltado principalmente para trabalhos universitários, com diferentes tamanhos de equipes, e pode ser usado por usuários com pouca experiência na área, com foco em manter a documentação bem estruturada e tentar fazer isso de forma ágil. O processo é baseado no trabalho de MARTINS (2001), que apresenta uma metodologia de elicitação de requisitos e o nível G do MPS .Br, que é onde se encontra o processo de engenharia de requisitos.

## 7 RESULTADOS OBTIDOS

Nesta sessão serão mostrados os resultados obtidos, através da aplicação do guia no N2S e na Startup Recicla Go. Onde o N2S, conta com 11 estagiários, onde na primeira Sprint os estagiários levaram mais de uma semana para realizar atividades simples como, o planejamento da Sprint e de estudo do sistema, graças a isso a Sprint #1 teve que ter um acréscimo de uma semana para a conclusão das atividades, após o treinamento feito com base no processo resultante da seleção das características do núcleo, processo pensado especificamente para essas características (Apêndice 3), as atividades de planejamentos se tornaram mais rápida, sendo realizadas em um dia, e os estagiários agora sabiam por onde começar outras atividades. No Recicla Go, existia um documento com a ideia de manual para o sistema, porém em quase 4 meses o desenvolvimento ficou parado, nenhum outro documento ou decisões foram tomadas, pois não havia um plano a ser seguido.

No N2S o processo foi iniciado no dia 03 de fevereiro de 2021, iniciando com o treinamento, antes desse período já tinha sido observado uma dificuldade dos estagiários atuais, graças a isso foi possível fazer uma experiência com os mesmos times de estagiários antes e após processo, o processo escolhido foi Modelo de Processo Ágil com Cliente, Apêndice 3, a seguir alguns pontos que foram observados antes do treinamento.

- **Documentação incompleta:** as documentações ficavam muito no básico do sistema, resultado em documentos que deixavam as equipes sem a visão completa do sistema.
- **Códigos sem práticas de programação adequadas:** os códigos eram feitos, porém não eram documentadas as escolhas ou nem mesmo comentados no código de forma a facilitar para os próximos times, isso fez os novos times perderem muito tempo tentando entender os códigos.
- **Falta de relatórios:** as equipes não faziam relatórios ou documentavam suas decisões, o que pode gerar complicações futuras, fazendo equipes terem retrabalhos, até mais de uma vez.
- **Entendimento do sistema:** os documentos mostravam uma coisa, mas não tinha como saber o que teria ou não sido feito no sistema.
- **Desatualização:** os documentos não estavam completamente atualizados, mostrando uma carência em detalhes relevantes para o sistema.

- **Pandemia:** com o início da pandemia, os estagiários tiveram que iniciar atividades remotas, mas o processo não estava bem definido, e com o estado dos documentos e dos códigos, foi perdido muito tempo só preparando os documentos e os códigos que deveriam estar prontos.
- **Drive:** o drive das equipes não estava completamente organizado, o que tornou o trabalho um pouco mais difícil, pois quando precisávamos procurar documentos de equipes anteriores, ou até mesmo entender em qual documento estava aquela informação não tínhamos uma maneira rápida, o que fazia ter uma grande quantidade de tempo gasto em somente procurar documentos.

Após o treinamento os estagiários conseguiram realizar as atividades da forma que os coordenadores esperavam, além de conseguir fazer em um tempo reduzido, a seguir a visão após o treinamento.

- **Comunicação:** a comunicação entre os estagiários continua, algo que vem de equipes anteriores e que se manteve no processo atual.
- **Implementação de Rastreamento:** a partir do novo processo foi criado um sistema de identificador para os documentos, assim sempre que precisar recorrer a informação de outros documentos só vai ser necessário conferir onde aquele documento faz referência na seção do documento referente a rastreamento.
- **Documentos:** as ideias das equipes foram trazidas como documentos que devem ser trabalhados durante o estágio para manter os documentos atualizados e mostrando exatamente como o sistema está.
- **Controle:** foi criado um processo somente para as mudanças, incluindo um documento, para que o responsável pela mudança possa documentar qual foi a mudança, porque ela foi feita, e os documentos impactados pela mudança, para que possa ser rastreada, incluindo uma justificativa da mudança.
- **Práticas de programação:** no processo recomendamos aos estagiários a utilização das práticas como, nomes intuitivos no código, comentar em partes do código como funções que não são tão claras, separar os códigos conforme padrões de projeto em vez de programar tudo em um mesmo lugar.
- **Testes:** agora os testes são feitos inteiramente com base nos casos de uso, e apresentados através de cores, onde cada cor representa um estado do sistema, como por exemplo, verde se a funcionalidade está completa e funcionando.
- **Treinamento:** com a necessidade, agora existe um treinamento que inclui o processo que os estagiários trabalham, onde todos podem acessar sempre que tiverem dúvidas.
- **Reunião com equipes anteriores:** com o novo processo as reuniões com times anteriores são muito mais estimuladas e já melhor trabalhada, onde são planejadas com antecedência e a pau-

ta da reunião deve ser passada a todos os participantes, com os aprendizados que foram obtidos no passado, incluindo agora o ritmo de reuniões através de videochamadas.

Porém os estagiários ainda possuem uma certa dificuldade com o rastreamento dos documentos, que ainda está sendo trabalhado. Após o treinamento e com três Sprints feitas com o processo em prática os estagiários já tiveram grandes avanços notáveis, incluindo na própria visão deles, a seguir os dados fornecidos pelos próprios estagiários.

- **Praticidade:** os estagiários em sua maioria acharam o processo de fácil compreensão e prático de implementar.
- **Modelos:** os estagiários acharam os modelos de documentos propostos fáceis de usar e que com os modelos é possível focar mais tempo em aplicar e fazer as tarefas do que planejando um documento.
- **Boas Práticas:** os estagiários começaram a usar algumas das boas práticas propostas pelo processo, tais como, programação em pares, comunicação frequente entre a equipe, a troca e armazenamento de conhecimento entre as equipes, a organização do ambiente de trabalho (Drive), trocas de experiência com times anteriores de seus projetos.

Já na Startup Recicla Go, o processo proposto pelo guia começou a ser implementado no dia 8 de janeiro de 2021 e o modelo utilizado foi o Modelo de Processo Ágil com cliente implícito. Antes do processo ser instalado a Startup não conseguia seguir com seu projeto de forma eficiente, estava estagnado, após a instalação do modelo proposto, em apenas dois meses o projeto já estava indo para sua fase de desenvolvimento, porém antes a Startup vinha enfrentado sérios problemas com planejamento e prazos, pois sem um modelo fixo, não conseguiam planejar o que deveria ser feito e passavam mais tempo refazendo do que avançando no desenvolvimento. Mas, diferentemente do N2S, mostrado anteriormente, os dados vêm proveniente apenas de dois responsáveis, uma vez que a Startup ainda não possui muitos membros e os que estão alocados para o processo são apenas o gerente e um supervisor. O que fez o processo progredir tão rapidamente, o treinamento foi feito enquanto já produzia os documentos e os resultados foram aprovados pelos 6 membros, incluindo os dois mencionados antes.

Ambos os processos trouxeram resultados positivos para os projetos, durante a experiência algumas alterações foram feitas no processo, principalmente no modelo usado no N2S.

## 8 CONCLUSÃO E TRABALHOS FUTUROS

Este trabalho teve como principal objetivo criar um guia para que usuários possam selecionar o processo de engenharia de requisito mais adequado conforme as características da equipe, empresa, Startup, núcleo ou grupo de forma rápida e prática. A elaboração do trabalho se deu a partir de pesquisas para identificar as características de cada perfil, onde foram feitas entrevistas, questionários e observações. Após a pesquisa, foi possível criar o guia. Em seguida, o guia foi validado de acordo com alguns perfis. A seguir será apresentado os trabalhos futuros encontrados para este projeto, durante as validações foram encontrados três trabalhos:

O primeiro é a adaptação de um processo para o formato remoto, a ideia surgiu devido ao estado de isolamento social vivenciado durante o período da pesquisa, esse processo seria completamente voltado para empresas que adotaram o *home office* e precisam de um novo processo adaptado, então um novo processo ágil será criado com base nos modelos ágeis que já foram propostos, sendo adaptado para o *home office*.

O segundo trabalho é a adaptação dos processos de engenharia de requisitos, para processo de desenvolvimento de jogos, em uma Startup que ainda está no processo de criação, voltada para o desenvolvimento de jogos, foi notado que os processos não são muito adequados, mas que podem ser adaptados para essa realidade.

Por fim o terceiro trabalho é a aplicação do guia no meio acadêmico, em sala de aula, um trabalho que seria realizado durante este trabalho de pesquisa, porém devido a complicações como, tempo, que não foi possível concluir o guia a tempo para ser usado em sala de aula, e o próprio formato *home office* de aulas que se tornou um obstáculo.

## REFERÊNCIAS

- ALENCAR, A. I. S. Criação de um Guia para selecionar métodos de avaliação para tecnologias de software M - Learning. 2019. 63F. Monografia (Graduação em Engenharia de Software) - Campus de Russas, Universidade Federal do Ceará, Russas, 2019.
- BORGES, E. P. DESENVOLVIMENTO ÁGIL EM NÚMEROS, NO BRASIL E NO MUNDO, 2017. Disponível em: <https://www.igti.com.br/blog/desenvolvimento-agil-em-numeros/>. Acesso em: 19 abr. 2020.
- CAVALCANTE, M. B. Proposta de um modelo baseado na combinação de metodologias ágeis de desenvolvimento de software. 2018. 115 f. Monografia (Graduação em Engenharia de Software) – Campus de Russas, Universidade Federal do Ceará, Russas, 2018.
- CINTRA, C. C. A implementação de um processo de engenharia de requisitos baseado no Processo Unificado da Rational (RUP) alcançando nível 3 de Maturidade da Integração de Modelos de Capacidade e Maturidade (CMMI) incluindo a utilização de práticas de métodos ágeis. 2006 – Universidade Federal do Rio Grande do Sul, Instituto de Informática, Programa de Pós-Graduação em Computação, Porto Alegre, RG. Disponível em: <https://www.lume.ufrgs.br/handle/10183/8128>. Acesso em 17 ago. 2020.
- LEANDRO. M. T. R. CMMI USAR OU NÃO USAR?. 2016. Disponível em: <http://leandromtr.com/cmmi-usar-ou-nao-usar/>. Acesso em: 27 ago. 2020.
- JAQUEIRA, A. O. P. Uso de modelos i\* para enriquecer requisitos em métodos ágeis. 2013. 97 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal do Rio Grande do Norte, Natal, 2013.
- ISDBRASIL. O QUE É CMMI?. Disponível em: <http://www.isdbrasil.com.br/o-que-e-cmmi.php>. Acesso em: 17 ago. 2020.
- MARTINS, L. E. G. Uma metodologia de elicitação de requisitos de software baseada na teoria da atividade. 2001. 170p. Tese (doutorado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação, Campinas, SP. Disponível em: <<http://www.repositorio.unicamp.br/handle/REPOSIP/260233>>. Acesso em: 17 ago. 2020.
- MPS.BR - Melhoria de Processo do Software Brasileiro: Guia Geral MPS de Software, SofTex, 2020. Disponível em: <https://softex.br/mpsbr/guias/#guia-sw>. Acesso em: 17 ago. 2020.
- POLCELLI, A. Um estudo sobre a relevância de boas práticas de engenharia de requisitos. 2005. 162p. Dissertação (mestrado profissional) - Universidade Estadual de Campinas, Instituto de Computação, Campinas, SP. Disponível em: <http://www.repositorio.unicamp.br/handle/REPOSIP/276471>. Acesso em: 17 ago. 2020.
- PRESSMAN, R. S. ENGENHARIA DE SOFTWARE: Uma Abordagem Profissional. 7ª ed. Dados eletrônicos. Porto Alegre: AMGH, 2011.
- SILVA, D. O que é o MPS.Br, 2013. Disponível em: <https://blogdaqualidade.com.br/o-que-e-o-mps-br/>. Acesso em: 27 ago. 2020.

SIMÕES, G. S. A suma importância da Engenharia de Requisitos. Revista FATTO, 2017. Disponível em: <http://www.fattocs.com/pt/artigo-a-suma-importancia-da-engenharia-de-requisitos>. Acesso em: 18 abr. 2020.

SOMMERVILLE, I. ENGENHARIA DE SOFTWARE. 9ª ed. Pearson Education do Brasil. São Paulo, 2011.

VAZQUEZ, C. E.; SIMÕES, G. S. ENGENHARIA DE REQUISITOS: Software Orientado ao Negócio. 1ª ed. Brasport Livros e Multimídia Ltda. Rio de Janeiro, 2016.

## APÊNDICES

### Apêndice 1 – Modelo de Estórias de Usuário com Cliente

# Modelo de Processo Ágil baseados em estórias de usuário com clientes

Modelo proposto por: Lucas Rafael Alves Oliveira

Março de 2021

## SUMÁRIO

**Introdução**     **3**

**Objetivo**       **3**

**Modelo Gráfico**     **3**

**Descrição das Atividades**     **4**

**Referências**     **6**

**Apêndices**       **7**

## 1. Introdução

Pensando em empresas que precisam de velocidade em seus processos, o modelo de processo ágil baseado em histórias de usuário foi criado. O modelo será feito de adaptações de modelos já existentes em conjunto com as atividades de Engenharia de requisitos dos modelos do CMMI e MPS Br, além de ser focado em empresas que desenvolvem diretamente para um cliente. O modelo proposto irá transformar os requisitos de usuários em histórias de usuário e a partir delas montar um modelo gráfico do sistema e seus requisitos. Esse documento está dividido em seis sessões, a 1ª constitui a introdução, a 2ª será mostrado os objetivos do modelo, a 3ª será apresentado o modelo gráfico do processo para facilitar a compreensão, na 4ª será explicado as atividades envolvidas mostradas no modelo gráfico, a 5ª serão apresentadas as referências para a construção do modelo e a 6ª serão os Apêndices, onde serão colocados exemplos de artefatos para conduzir o treinamento dos usuários.

## 2. Objetivo

O processo tem como objetivo dar aos usuários do guia um processo ágil e bem simplificado, mas que ainda assim possua os quesitos de qualidade proposto pelo CMMI e MPS Br. Assim fornecendo agilidade, facilidade e qualidade.

## 3. Modelo Gráfico

Nesta sessão iremos ver o modelo gráfico do processo de transformação de Histórias de Usuário em diagramas, no modelo que é mostrado na Figura 1 e na Figura 2 será utilizado o diagrama de casos de uso.

Figura 1 - Processo de Histórias de Usuários.

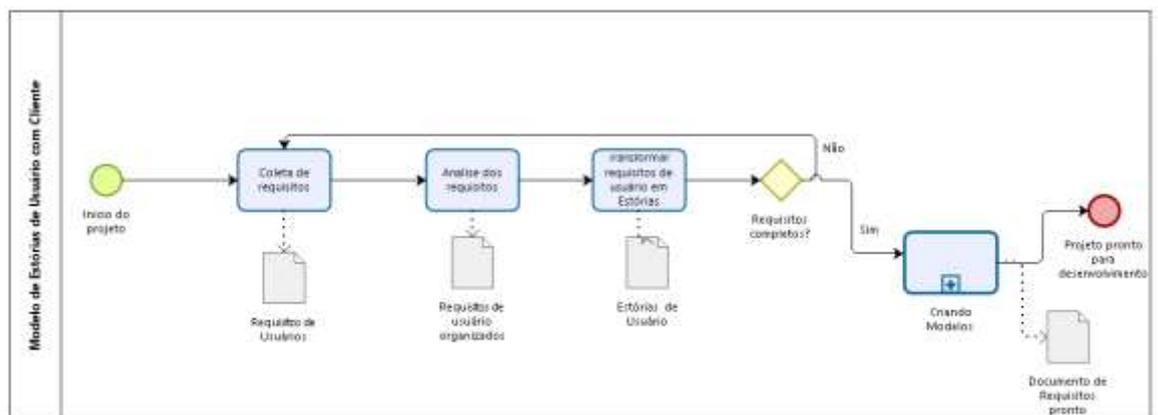
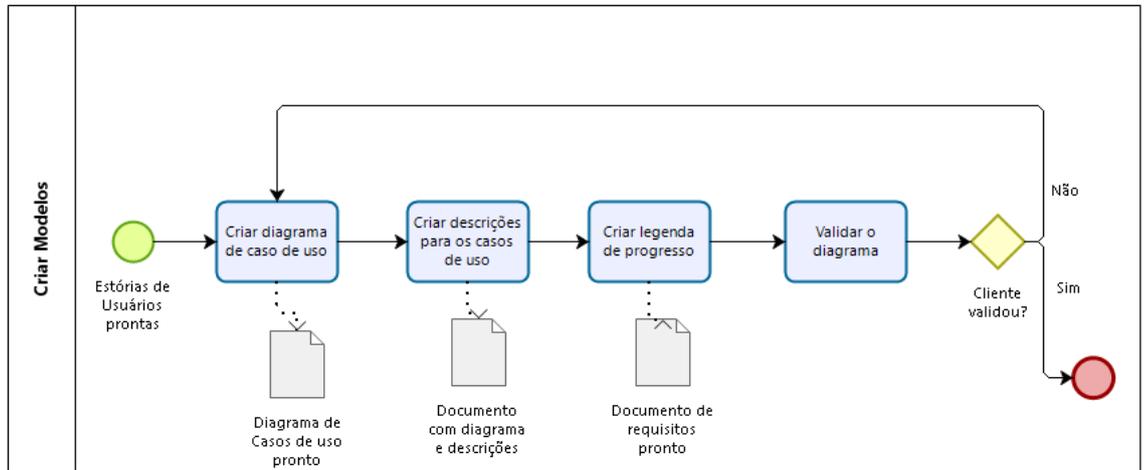


Figura 2 - Subprocesso de Estórias de Usuário.



Fonte: Autor do trabalho.

#### 4. Descrição das Atividades

Nesta sessão veremos a descrição das atividades, eventos e artefatos mostrados nos processos mostrados na Figura 1 e na Figura 2, será detalhado na mesma ordem de apresentação do modelo, mostrando em uma lista, além de incluir ferramentas que podem ser úteis no processo. Cada atividade, artefato ou evento mostrado anteriormente será listado e estarão destacados no texto em negrito.

- **Início do projeto:** O evento de start do processo, ocorre quando o cliente entrar em contato com a equipe, solicitando o desenvolvimento de um sistema.
- **Coleta de requisitos:** A atividade de coleta de requisitos é realizada logo no início do projeto, onde um ou mais membros da equipe devem fazer uma ou mais reuniões com o cliente, realizando entrevista, questionário e/ou observação, a realização de uma não causa a exclusão de outra, podem ser usadas as três, depende muito da disponibilidade do cliente e dos futuros usuários, com os resultados da entrevista, questionário e/ou observação a equipe terá em mão os requisitos de usuário, que devem estar documentados.
- **Requisitos de Usuários:** Documento que contém os resultados feito pela atividade de coleta de requisitos, requisitos informais e desorganizados, porém são a base das estórias de usuário, qualquer requisito que pode ser mal interpretado futuramente deve ser esclarecido o quanto antes com os clientes e usuários.
- **Análise dos Requisitos:** Nesta atividade a equipe deve documentar os requisitos de usuário, de forma a remover os ambíguos, as repetições e os ilegíveis, porém ainda será mantido em formato de requisitos de usuário, apenas será feito revisões ortográficas e deixando-os mais legíveis e já colocando níveis de prioridade.
- **Requisitos de Usuários Organizados:** No final da tarefa os requisitos de usuários estarão todos em um documento, agora com as prioridades pré estabelecidas e organizados em ordem de preferência dos desenvolvedores, recomendado que sejam feitos na ordem de

funcionalidades parecidas ou em dependência de funcionalidade, onde a funcionalidade que depende de outra funcionalidade vem logo após a sua dependência.

- **Transformar requisitos de usuários em histórias de usuários:** Neste ponto os desenvolvedores devem usar os requisitos coletados para fazer as histórias de usuário, onde serão separados cada funcionalidade em um cartão, onde no cartão conterà o identificador da história, o nome da história, os atores que vão realizar aquela funcionalidade, a descrição da interação dos atores com a funcionalidade, os requisitos envolvidos, as condições para aquela história ocorrer e a prioridade.
- **Histórias de Usuário:** Um documento com os cartões de histórias de usuário, esse mesmo documento deve conter um sumário, um glossário para os termos e abreviações, uma introdução, pois ele se tornará posteriormente o documento de requisitos.
- **Requisitos estão prontos?:** Após finalizar a atividade, os requisitos e histórias de usuários devem ser validadas junto ao cliente ou usuários, caso eles estejam de acordo o processo deve continuar, caso não atualize os documentos com base nos apontamentos sugeridos pelos usuários ou cliente, caso ache necessário refaça a atividade de coletas de requisitos, caso encontre algum requisito que foi deixado de fora da primeira vez, lembre-se que os requisitos podem mudar constantemente.
- **Criando Modelos:** Quando os requisitos estiverem prontos e validados, os desenvolvedores devem começar a fase de criação dos modelos gráficos do sistema.
  - **Histórias de usuários prontas:** Esse evento marca o início da fase de criação do modelo gráfico, assim que as histórias estiverem prontas e validadas o processo está apto a entrar na fase de construção dos diagramas, será focado no de caso de uso, porém se a equipe achar necessário eles podem utilizar os demais diagramas disponíveis, como por exemplo o de classe.
  - **Criar diagramas de caso de uso:** Com base nas histórias de usuários os desenvolvedores devem criar o diagrama de casos de uso, onde deve ser mostrado os atores (Usuários do sistema) interagindo com as funcionalidades, lembre-se de manter o diagrama simples, quanto mais poluído ele estiver mais difícil será para o cliente entender.
  - **Diagrama de casos de uso pronto:** Após a tarefa o diagrama gerado é colocado em uma sessão do documento de requisitos destinada especificamente ao diagrama.
  - **Criar descrições para os casos de uso:** Agora para cada caso de uso que está no diagrama, deve ser feito um quadro com o identificador do caso de uso, com o nome do caso de uso, com os atores, com uma descrição, com as histórias de usuário envolvidas, com a pré-condição para realizar a atividade, com o resultado que deve ter sido obtido pelo usuário, com o fluxo, principal, fluxo alternativo e de exceção.
  - **Documento com diagrama e descrições:** Documento de requisitos com a sessão de descrição dos casos de uso.
  - **Criar legendas de progresso:** Nesta atividade uma legenda deve ser adicionada para manter o controle do que já foi desenvolvido, o que está sendo desenvolvido e o que ainda deve ser desenvolvido, pode ser usado por exemplo um esquema de cores para representar.
  - **Documento de requisitos pronto:** Agora os desenvolvedores têm a primeira versão do documento de requisitos pronto, lembre-se de manter um controle de versão para as futuras alterações.

- **O cliente validou?** Caso o cliente aponte alguma alteração, refaça as atividades do subprocesso com as alterações solicitadas, caso ele aprove a equipe terá o documento pronto.
- **Projeto pronto para o desenvolvimento:** Com isso o projeto já está pronto para o desenvolvimento do sistema, porém quando houver mudanças a fase de engenharia de requisitos mostrado nesse processo pode ser refeita.

Para esse processo são recomendados usar o trello, uma ferramenta de gestão de tarefas, assim a equipe saberá o que tem que fazer e o que cada um já tem, e o que estão fazendo, é bem simples de se usar e basta ter uma conta no google para usar.

Outra ferramenta são as disponibilizadas pelo google drive, como o docs, assim todos terão acesso aos documentos mais rapidamente, além de poder fazer apresentações e ter planilhas para o uso da equipe quando necessário.

As entrevistas, questionários e observações ficam a critério dos desenvolvedores, pois eles variam facilmente conforme o sistema solicitado, porém ao final deste documento terá em Apêndice um exemplo de um documento de requisito completo usado por este processo.

## 5. Referências

JAQUEIRA, Aline de Oliveira Prata. Uso de modelos i\* para enriquecer requisitos em métodos ágeis. 2013. 97 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal do Rio Grande do Norte, Natal, 2013.

MARTINS, Luiz Eduardo Galvão. Uma metodologia de elicitação de requisitos de software baseada na teoria da atividade. 2001. 170p. Tese (doutorado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação, Campinas, SP. Disponível em: <<http://www.repositorio.unicamp.br/handle/REPOSIP/260233>>. Acesso em: 17 ago. 2020.

MPS.BR - Melhoria de Processo do Software Brasileiro: Guia Geral MPS de Software, SofTex, 2020. Disponível em: <<https://softex.br/mpsbr/guias/#guia-sw>>. Acesso em: 17 ago. 2020.

ISDBRASIL. O QUE É CMMI?. Disponível em: <<http://www.isdbrazil.com.br/o-que-e-cmmi.php>>. Acesso em: 17 ago. 2020.

6. Apêndices

**APÊNDICE 1 - Exemplo de documento de requisitos**

(Imagem representando a empresa)

## Documento de Requisitos

### Sistema de vendas de ingressos de um cinema

**INTRODUÇÃO** 2

**GLOSSÁRIO** 2

**REQUISITOS DE USUÁRIO** 2

**ESTÓRIAS DE USUÁRIO** 3

**DIAGRAMA DE CASOS DE USO** 9

**DESCRIÇÕES DE CASOS DE USO** 9

<b>Versão</b>	<b>Data</b>	<b>Autor</b>	<b>Descrição</b>
1.0	17/11/2020	Lucas Rafael	Criação do documento, adição da introdução, do glossário, dos requisitos de usuário, das estórias de usuário, do diagrama de casos de uso e das descrições de caso de uso.

## 1. INTRODUÇÃO

O sistema a ser desenvolvido compreende a venda de ingressos de um cinema, de forma a ajudar os funcionários e tornar todo o processo mais rápido e fácil, além de contribuir a manter o controle das vendas, e este documento é dividido em introdução, glossário, onde serão apresentados os termos, convenções, abreviaturas e os identificadores, em seguida será os requisitos de usuário, logo em seguida as estórias de usuário, o diagrama de caso de uso e a descrição de caso de uso.

## 2. GLOSSÁRIO

Nesta seção veremos os termos, abreviações, convenções e identificadores:

Identificadores: Requisitos de Usuário(RS001), Estórias de Usuário(EU001), Casos de Uso(UC001).

Abreviações: ID(Identificador).

## 3. REQUISITOS DE USUÁRIO

Nesta seção veremos os requisitos coletados com o cliente e os usuários, os requisitos serão divididos por funcionalidades e cada um terá sua prioridade.

<b>ID</b>	<b>Descrição</b>	<b>Prioridade</b>
RS001	Quero que o sistema possa ser acessado por mim e os meus funcionários, quero que possamos modificar os dados e cada um pode excluir seus cadastros, mas só eu posso excluir todos.	Essencial

RS002	Quero ter acesso ao balanceamento financeiro do dia e o mensal.	Essencial
RS003	Quero que mostre quantos assentos tem disponível na hora de vender.	Essencial
RS004	Quero que os preços possam ser definidos e sejam mostrados na hora da compra.	Essencial
RS005	Quero que gere o papel com as informações do ingresso, como hora da sessão, preço do ingresso, assento e sala.	Essencial
RS006	Quero ter acesso aos históricos de finanças e relatórios diários e mensais.	Essencial
RS007	Quero que seja possível criar novas sessões de filmes, além de excluir e modificá-la.	Essencial

#### 4. ESTÓRIAS DE USUÁRIO

<b>ID: EU001</b>
<b>Título: “Cadastro no sistema”</b>
<b>Ator:</b> Gerente, funcionário.
Como Gerente/Funcionário quero ser capaz de fazer meu cadastro no sistema.

<b>Prioridade:</b> Essencial
<p><b>Cr�terios de Aceita�o:</b></p> <p>1 - O sistema deve apresentar uma tela para o usu�rio poder realizar seu cadastro e ser capaz de inserir o nome, e-mail, senha e cpf.</p> <p>2 - Quando o usu�rio preencher todos os dados, o sistema deve verificar se o cpf � v�lido.</p> <p>3 - Quando selecionado a op�o "Cadastrar" o sistema deve cadastrar os dados do usu�rio no sistema.</p>
<b>Requisitos:</b> RS001

<b>ID:</b> EU002
<b>T�tulo:</b> "login no sistema"
<p><b>Ator:</b> Gerente, funcion�rio.</p> <p>Como Gerente/Funcion�rio quero ser capaz de fazer login e acessar as funcionalidades do sistema.</p> <p><b>Prioridade:</b> Essencial</p>
<p><b>Cr�terios de Aceita�o:</b></p> <p>1 - O sistema deve apresentar uma tela para o usu�rio inserir seus dados de login como e-</p>

mail e senha.

2 - Quando o usuário preencher os campos de e-mail e senha e clicar no botão de entrar o sistema deve verificar se existe no banco de dados.

3 - Caso exista no banco de dados o sistema efetua o login, caso contrário avisa que o usuário não possui cadastro.

**Requisitos: RS001**

**ID: EU003**

**Título: “Modificar Cadastro”**

**Ator:** Gerente, funcionário.

Como Gerente/Funcionário quero ser capaz de fazer alterações no meu cadastro no sistema.

**Prioridade:** Essencial

**Crterios de Aceitaço:**

1 - Depois de já estar logado no sistema o usuário pode clicar no ícone de perfil e lá deve ser capaz de fazer alterações nos seus dados cadastrais.

2 - Quando o usuário finalizar as alterações o sistema deve verificar se as alterações são válidas.

3 - Quando selecionado a opção “Salvar” o sistema deve alterar a as informações no banco de dados.

**Requisitos: RS001**

**ID: EU004**

**Título: “Excluir cadastro no sistema”**

**Ator:** Gerente, funcionário.

Como Gerente/Funcionário quero ser capaz de excluir meu cadastro no sistema.

**Prioridade:** Essencial

**Critérios de Aceitação:**

1 - Com o usuário logado no sistema, deva ser capaz de excluir o seu próprio perfil, somente o gerente deve ser capaz de excluir os perfis dos funcionários.

**Requisitos: RS001**

**ID: EU005**

**Título: “Acessar Relatórios”**

**Ator:** Gerente.

Como Gerente quero ser capaz de acessar relatórios mensais e diários informando os balanços financeiros das vendas.

**Prioridade:** Essencial

**Critérios de Aceitação:**

- 1 - O sistema deve apresentar uma tela para o usuário acessar os relatórios gerados.
- 2 - Quando o usuário acessar a página de relatórios o sistema deve gerar um PDF mostrando o histórico de vendas do dia.
- 3 - Quando o usuário selecionar a opção “Relatório Mensal” o sistema deve gerar um PDF com o histórico financeiro do mês em questão.

**Requisitos:** RS002

**ID:** EU006

**Título:** “Verificar disponibilidade”

**Ator:** Funcionário.

Como Funcionário quero ser capaz de ver quantos assentos ainda tem disponível para determinado filme e sessão.

<b>Prioridade:</b> Essencial
<b>Critérios de Aceitação:</b>  1 - Quando o Usuário acessar a parte de venda de um determinado filme e sessão o sistema deve mostrar quantos acentos ainda tem disponível.
<b>Requisitos:</b> RS003

<b>ID:</b> EU007
<b>Título:</b> “Vender Ingressos”  <b>Ator:</b> Funcionário.  Como Funcionário quero ser capaz de definir os preços dos ingressos e quando for para vender eu possa escolher entre os preços de um ingresso inteiro ou um ingresso com desconto. <b>Prioridade:</b> Essencial
<b>Critérios de Aceitação:</b>  1 - O sistema deve permitir que o usuário seja capaz de definir os preços toda vez que uma nova sessão seja criada.  2 - O sistema deve apresentar os preços definidos pelo usuário quando ele for vender um ingresso.

**Requisitos: RS004**

**ID: EU008**

**Título: “Gerar Ingresso”**

**Ator:** Funcionário.

Como Funcionário quero que o sistema seja capaz de gerar um ingresso quando eu confirmar a venda.

**Prioridade:** Essencial

**Crterios de Aceitaço:**

1 - O sistema deve ser capaz de gerar um ingresso com as informaes de filmes, preo e sesso aps o usurio confirmar a venda.

**Requisitos: RS005**

**ID: EU009**

**Título: “Visualizar Histrico”**

**Ator:** Gerente.

Como Gerente quero ser capaz de visualizar o histórico financeiro dos meses anteriores.

**Prioridade:** Essencial

**Critérios de Aceitação:**

1 - O sistema deve apresentar a opção de gerar um histórico dos meses anteriores.

**Requisitos:** RS006

**ID:** EU010

**Título:** “Cadastrar sessão”

**Ator:** Gerente, funcionário.

Como Gerente/Funcionário quero ser capaz de cadastrar uma nova sessão no sistema.

**Prioridade:** Essencial

**Critérios de Aceitação:**

1 - O sistema deve permitir que os usuários que estejam logados cadastrem novas sessões de filmes, com o título do filme, duração, hora de início da sessão, tempo em cartaz, valor do ingresso e sala da sessão.

2 - Quando o usuário selecionar a opção de “Salvar” o sistema deve verificar se as informações são válidas.

3 - Se as informações forem válidas o sistema deve salvar a nova sessão no banco de dados, caso contrário o sistema deve informar que não foi possível concluir o cadastro por informações inválidas.

**Requisitos: RS007**

**ID: EU011**

**Título: “Modificar Sessão”**

**Ator:** Gerente, funcionário.

Como Gerente/Funcionário quero ser capaz de modificar as sessões existentes no sistema. **Prioridade:** Essencial

**Crterios de Aceitação:**

1 - Com o usuário logado no sistema, deve ser capaz de modificar os dados de uma determinada sessão.

2 - Quando o usuário selecionar a opção de “Salvar” as modificações, o sistema deve verificar se as modificações são válidas.

3 - Se as modificações forem válidas o sistema deve salvar as modificações no banco de

dados, caso contrário deve informar ao usuário que existem informações inválidas.

**Requisitos: RS007**

**ID: EU012**

**Título: “Excluir Sessão”**

**Ator:** Gerente, funcionário.

Como Gerente/Funcionário quero ser capaz de excluir uma sessão existente do sistema.

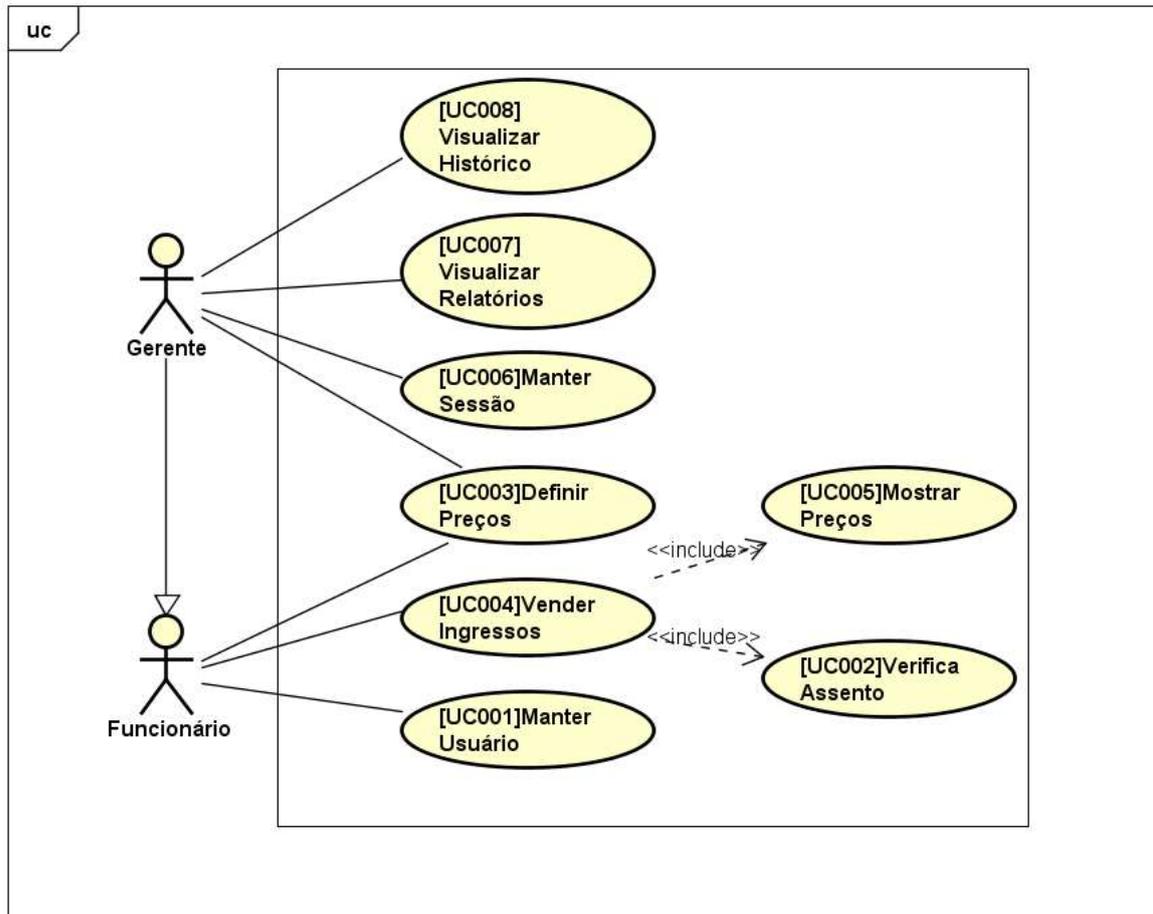
**Prioridade:** Essencial

**Critérios de Aceitação:**

- 1 - O usuário deve estar logado no sistema para conseguir excluir uma sessão.
- 2 - O usuário deve selecionar a opção “Excluir” e o sistema deve remover a sessão do banco de dados.

**Requisitos: RS007**

## 5. DIAGRAMA DE CASOS DE USO



powered by Astah

## 6. DESCRIÇÕES DE CASOS DE USO

Essa seção destina-se a realizar a descrição detalhada dos casos de uso identificados no sistema de venda do sistema. As caixas de texto são divididas em cinco categorias que dizem respeito ao andamento do desenvolvimento dos componentes de venda do cinema. O nível de desenvolvimento do componente é exibido na legenda abaixo:



**Componente desenvolvido**



**Componente em desenvolvimento****Componente desenvolvido, mas apresentou falhas****Componente ainda não desenvolvido****Situação não identificada**

Lista de casos de uso descritos:

<b>ID</b>	<b>UC001</b>
Nome	Manter Usuário
Ator Primário	Gerente, Funcionário
Sumário	Este caso de uso tem como objetivo mostrar o passo a passo que o usuário deve fazer para se cadastrar, logar, modificar e excluir.
Estória de Usuário	[EU001], [EU002], [EU003] e [EU004]
Pré-condições	O usuário ter, ou fazer um novo cadastro.

Pós-condições	Um perfil novo cadastrado, modificado, excluído ou logado
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O sistema apresenta os ícones de “entrar” ou fazer “novo cadastro”.</li> <li>2. O usuário seleciona a opção de “entrar”.</li> <li>3. O sistema exibe a página de login, com o campo “e-mail”, “senha” e os botões de “Entrar” e “Criar novo”</li> <li>4. O usuário insere os dados e seleciona em “Entrar”.</li> <li>5. O sistema verifica a existência do usuário e carrega a página inicial.</li> <li>6. O caso de uso é encerrado.</li> </ol>
Fluxo Alternativo	<p>A2.1 - O usuário seleciona a opção de “novo cadastro”.</p> <p>A2.2 - O sistema exibe uma página com os campos, “nome”, “e-mail”, “cpf”, “cargo” e os botões de "Criar" e “Entrar”.</p> <p>A2.3 - O usuário preenche os dados e seleciona a opção “Criar”.</p> <p>A2.4 - O sistema cria o usuário no banco de dados.</p> <p>A2.5 - O caso de uso retorna para o fluxo principal no passo 5.</p> <p>A3.1 - O usuário seleciona a opção “Criar novo”.</p> <p>A3.2 - O caso de uso volta para o fluxo alternativo A2.</p> <p>A5.1 - O sistema não encontra o usuário.</p> <p>A5.2 - O sistema notifica que o usuário não foi encontrado.</p> <p>A5.3 - O caso de uso retorna ao fluxo principal no passo 3.</p>

Fluxo de Exceção	EA2.2.1 - O usuário seleciona a opção “Entrar”. EA2.2.2 - O caso de uso retorna ao fluxo principal no passo 3.
------------------	---

ID	UC002
Nome	Verificar Assento
Ator Primário	Realizada ou vender ingresso
Sumário	Este caso de uso tem como objetivo mostrar como deve ser o passo a passo que o sistema deve realizar para verificar os assentos do cinema
Estórias de Usuários	[EU006]
Pré-condições	O usuário deve estar logado e efetuando a venda de um ingresso
Pós-condições	O usuário deve saber quais assentos estão disponíveis
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O usuário acessa o menu de vendas.</li> <li>2. O sistema apresenta a tela pedindo as informações da sessão.</li> <li>3. O usuário preenche as informações.</li> <li>4. O sistema verifica através de uma tabela quais assentos ainda estão disponíveis</li> <li>5. O sistema apresenta uma lista de assentos vagos.</li> </ol>

	<p>6. O usuário seleciona um assento.</p> <p>7. O sistema salva a informação.</p> <p>8. O caso de uso é encerrado.</p>
Fluxo Alternativo	<p>A4.1 - O sistema verifica através de uma tabela quais assentos ainda estão disponíveis.</p> <p>A4.2 - O sistema indica que não há mais assentos vagos.</p> <p>A4.3 - O caso de uso retorna ao fluxo principal no passo 7.</p>
Fluxo de Exceção	

<b>ID</b>	<b>UC003</b>
Nome	Definir Preços
Ator Primário	Vendedor, Gerente
Sumário	Este caso de uso tem como objetivo mostrar o passo a passo que o usuário deve fazer para definir os preços dos ingressos.
Estórias de Usuários	[EU007]
Pré-condições	O usuário está previamente logado no sistema e existe pelo menos uma

	sessão cadastrada.
Pós-condições	Preços definidos
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O usuário seleciona a opção “Sessão”</li> <li>2. O sistema exibe um submenu.</li> <li>3. O usuário seleciona a opção “Definir preço”</li> <li>4. O sistema apresenta os campos de preços para os preços e a opção “definir”.</li> <li>5. O usuário preenche os campos de preço "inteiro" e “meia” e seleciona a opção de “definir”</li> <li>6. O sistema salva as informações de preços para aquela sessão.</li> <li>7. O sistema sugere para o usuário se ele quer definir os mesmos preços para todas as sessões e as opções “sim” e “não”.</li> <li>8. O usuário seleciona “sim”</li> <li>9. O sistema retorna à página inicial.</li> <li>10. O caso de uso é encerrado.</li> </ol>
Fluxo Alternativo	<p>A7.1 - O sistema sugere que os preços sejam os mesmos para todas as sessões e as opções “sim” e “não”.</p> <p>A7.2 - O usuário seleciona a opção “não”.</p> <p>A7.3 - O caso de uso retorna ao fluxo principal no passo 9.</p>
Fluxo de Exceção	

ID	UC004
Nome	Vender Ingresso
Ator Primário	Funcionário, Gerente
Sumário	Este caso de uso tem como objetivo mostrar o passo a passo que o usuário deve fazer para executar a funcionalidade de vender ingressos
Estória de Usuário	[EU007], [EU008]
Pré-condições	Usuário está cadastrado e logado no sistema
Pós-condições	Pelo menos um ingresso ter sido vendido
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O usuário acessa o menu venda.</li> <li>2. O sistema solicita que o usuário indique a sessão do ingresso.</li> <li>3. O usuário indica a sessão.</li> <li>4. O sistema apresenta os preços que o usuário pode selecionar e a opção de definir um novo preço, conforme o caso de uso [UC003] e [UC005].</li> <li>5. O usuário seleciona um dos preços existentes.</li> <li>6. O sistema mostra os assentos disponíveis conforme o caso de [UC002].</li> <li>7. O usuário seleciona o assento escolhido e confirma a venda.</li> <li>8. O sistema gera um ingresso com as informações da sessão,</li> </ol>

	<p>(nome do filme, horário, sala e preço).</p> <p>9. O sistema retorna à página inicial.</p> <p>10. O caso de uso é encerrado.</p>
Fluxo Alternativo	<p>A4.1 - O usuário seleciona a opção de definir um novo preço.</p> <p>A4.2 - O sistema realiza os passos do caso de uso [UC003].</p> <p>A4.3 - O caso de uso retorna ao fluxo principal no passo 5.</p> <p>A7.1 - O sistema não encontra assentos disponíveis e executa os passos conforme o fluxo alternativo do caso de uso [UC003].</p>
Fluxo de Exceção	

<b>ID</b>	<b>UC005</b>
Nome	Mostrar preços
Ator Primário	Realizado ao vender ingresso.
Sumário	Este caso de uso tem como objetivo mostrar o passo a passo que o sistema deve fazer para mostrar os preços dos ingressos da sessão selecionada para que o usuário possa marcar o preço desejado
Estórias de	[UC007].

Usuário	
Pré-condições	O usuário está cadastrado, logado e efetuando uma venda de ingresso.
Pós-condições	Um dos preços escolhidos ou novos preços definidos.
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O usuário está no processo de venda de um ingresso e seleciona a caixa de texto reservada para preço.</li> <li>2. O sistema verifica se existem preços definidos na sessão selecionada e apresenta os dois preços e a opção de definir novos preços.</li> <li>3. O usuário seleciona um dos preços.</li> <li>4. O caso de uso é encerrado.</li> </ol>
Fluxo Alternativo	<p>A2.1 - O sistema não identifica nenhum preço definido na sessão.</p> <p>A2.2 - O sistema solicita que o usuário defina os preços dos ingressos.</p> <p>A2.3 - O usuário seleciona a opção de definir preços e preenche os valores de “Inteira” e “Meia” dos ingressos e seleciona a opção “Confirmar”.</p> <p>A2.4 - O caso de uso retorna ao fluxo principal no passo 3.</p> <p>A3.1 - O usuário seleciona a opção de definir novos preços.</p> <p>A3.2 - O sistema exibe caixas de texto com os identificadores “Inteira” e “Meia” e a opção “Confirmar”.</p> <p>A3.3 - O usuário preenche os campos e seleciona a opção de “Confirmar”.</p> <p>A3.4 - O sistema salva as informações dos novos preços na sessão e pergunta ao usuário se os novos preços devem ser aplicados nas demais</p>

	<p>sessões, junto com as opções “Sim” e “Não”.</p> <p>A3.5 - O usuário seleciona a opção “Sim”.</p> <p>A3.6 - O sistema altera todas as sessões existentes para ter os mesmos valores de ingressos.</p> <p>A3.7 - O caso de uso retorna ao fluxo principal no passo 4.</p>
Fluxo de Exceção	<p>EA3.4.1 - O sistema salva as informações dos novos preços na sessão e pergunta ao usuário se os novos preços devem ser aplicados nas demais sessões, junto com as opções “Sim” e “Não”.</p> <p>EA3.4.2 - O usuário seleciona a opção “Não”.</p> <p>EA3.4.3 - O caso de uso retorna ao fluxo principal no passo 4.</p>

<b>ID</b>	<b>UC006</b>
Nome	Manter Sessão
Ator Primário	Gerente
Sumário	Este caso de uso tem como objetivo mostrar o passo a passo que o usuário deve fazer para cadastrar, modificar e excluir uma sessão.
Estórias de Usuários	[UC010], [UC011] e [UC012].
Pré-condições	O usuário deve estar previamente cadastrado e logado no sistema.

Pós-condições	Uma sessão deve ter sido cadastrada, modificada ou excluída com sucesso.
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O usuário vai até o menu sessão.</li> <li>2. O sistema apresenta um submenu: Cadastrar, Editar e Excluir.</li> <li>3. O usuário seleciona a opção de cadastrar.</li> <li>4. O sistema exibe uma página com um formulário com os campos, Filme, Horário, Sala, Qtdd de Assentos e Duração e a opção de “Salvar”.</li> <li>5. O usuário preenche as informações e seleciona a opção “Salvar”.</li> <li>6. O sistema salva as informações e cria uma tabela de assentos para a sessão.</li> <li>7. O sistema retorna à página inicial.</li> <li>8. O caso de uso é encerrado.</li> </ol>
Fluxo Alternativo	<p>A2.1.1 - O usuário seleciona a opção editar.</p> <p>A2.1.2 - O sistema abre uma página com as sessões existentes e a opção de “Editar” ao lado de cada uma.</p> <p>A2.1.3 - O usuário seleciona uma das sessões e seleciona a opção de “Editar” aquela sessão.</p> <p>A2.1.4 - O sistema exibe o formulário com os dados da sessão, Filme, Horário, Sala, Qtdd de Assentos, Duração e as opções “Salvar” e “Cancelar”.</p> <p>A2.1.5 - O usuário modifica as informações necessárias e seleciona a opção “Salvar”.</p> <p>A2.1.6 - O sistema salva as novas informações da sessão.</p> <p>A2.1.7 - O caso de uso retorna ao fluxo principal no passo 7.</p>

	<p>A2.2.1 - O usuário seleciona a opção excluir.</p> <p>A2.2.2 - O sistema abre uma página com as sessões existentes e a opção de “Excluir” ao lado de cada uma.</p> <p>A2.2.3 - O usuário seleciona uma das sessões e seleciona a opção de “Excluir” aquela sessão.</p> <p>A2.2.4 - O sistema pergunta se o usuário tem certeza se quer excluir a sessão e as opções “Sim” e “Não”.</p> <p>A2.2.5 - O usuário seleciona a opção “Sim”.</p> <p>A2.2.6 - O sistema remove as informações da sessão do banco de dados.</p> <p>A2.2.7 - O caso de uso retorna ao fluxo principal no passo 7.</p>
Fluxo de Exceção	<p>EA2.1.5.1 - O usuário seleciona a opção “Cancelar”.</p> <p>EA2.1.5.2 - O sistema retorna à página com as sessões existentes.</p> <p>EA2.1.5.3 - O caso de uso é encerrado.</p> <p>EA2.2.5.1 - O usuário seleciona a opção “Não”.</p> <p>EA2.2.5.2 - O sistema retorna à página com as sessões existentes.</p> <p>EA2.2.5.3 - O caso de uso é encerrado.</p>

<b>ID</b>	<b>UC007</b>
Nome	Visualizar Relatórios
Ator Primário	Gerente

Sumário	Este caso de uso tem como objetivo mostrar o passo a passo que o usuário deve fazer para visualizar os relatórios de venda do cinema.
Estórias de Usuários	[EU005].
Pré-condições	O usuário deve estar previamente cadastrado e logado no sistema.
Pós-condições	Ter pelo menos um relatório gerado.
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O usuário seleciona o menu relatório.</li> <li>2. O sistema gera uma página em PDF com o relatório do dia, contendo as informações de vendas e a opção de “Baixar Relatório” e “Gerar Relatório Mensal”.</li> <li>3. O usuário seleciona a opção de “Baixar relatório”.</li> <li>4. O sistema gera um arquivo PDF com o relatório do dia.</li> <li>5. O caso de uso é encerrado.</li> </ol>
Fluxo Alternativo	<p>A3.1 - O usuário seleciona a opção de “Gerar Relatório Mensal”.</p> <p>A3.2 - O sistema exibe uma página em PDF com o relatório de vendas de todos os dias daquele mês em questão e a soma do valor bruto total de venda do mês e a opção de “Baixar PDF”.</p> <p>A3.3 - O usuário seleciona a opção de “Baixar PDF”.</p> <p>A3.4 - O sistema gera um arquivo PDF com o relatório do mês.</p> <p>A3.5 - O caso de uso é encerrado.</p>
Fluxo de	

Exceção	
---------	--

<b>ID</b>	<b>UC008</b>
Nome	Visualizar Histórico.
Ator Primário	Gerente.
Sumário	Este caso de uso tem como objetivo mostrar o passo a passo que o usuário deve fazer para visualizar os históricos de vendas.
Estórias de Usuários	[EU009]
Pré-condições	O usuário deve estar previamente cadastrado e logado no sistema.
Pós-condições	O usuário deve ter conseguido ter acesso ao histórico de vendas.
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O usuário seleciona o menu de Histórico.</li> <li>2. O sistema apresenta um calendário para que o usuário selecione o período de início e de fim, para que o sistema mostre as informações do período.</li> <li>3. O usuário seleciona o período.</li> <li>4. O sistema exibe um PDF com as informações de venda dos dias do período e do total de cada mês do período e as opções “Baixar PDF” e “Verificar outro período”.</li> <li>5. O usuário seleciona a opção de “Baixar PDF”.</li> </ol>

	<p>6. O sistema gera um arquivo em PDF com o histórico do período.</p> <p>7. O caso de uso é encerrado.</p>
Fluxo Alternativo	<p>A5.1 - O usuário seleciona a opção de “Verificar outro período”.</p> <p>A5.2 - O caso de uso retorna ao fluxo principal no passo 2.</p>
Fluxo de Exceção	



**Apêndice 2 – Modelo de Processo de Estórias de Usuário com cliente implícito.**

# Modelo de Processo Ágil baseados em estórias de usuário com Cliente Implícito

Modelo proposto por: Lucas Rafael Alves Oliveira

Março de 2021

## SUMÁRIO

**Introdução**     **3**

**Objetivo**       **3**

**Modelo Gráfico**     **3**

**Descrição das Atividades**     **4**

**Referências**     **6**

**Apêndices**       **7**

## 1. Introdução

Pensando em empresas que precisam de velocidade em seus processos, o modelo de processo ágil baseado em histórias de usuário foi criado. O modelo será feito de adaptações de modelos já existentes em conjunto com as atividades de Engenharia de requisitos dos modelos do CMMI e MPS Br, além de ser focado em empresas que desenvolvem diretamente para um cliente. O modelo proposto irá transformar os requisitos de usuários em histórias de usuário e a partir delas montar um modelo gráfico do sistema e seus requisitos. Esse documento está dividido em seis sessões, a 1ª constitui a introdução, a 2ª será mostrado os objetivos do modelo, a 3ª será apresentado o modelo gráfico do processo para facilitar a compreensão, na 4ª será explicado as atividades envolvidas mostradas no modelo gráfico, a 5ª serão apresentadas as referências para a construção do modelo e a 6ª serão os Apêndices, onde serão colocados exemplos de artefatos para conduzir o treinamento dos usuários.

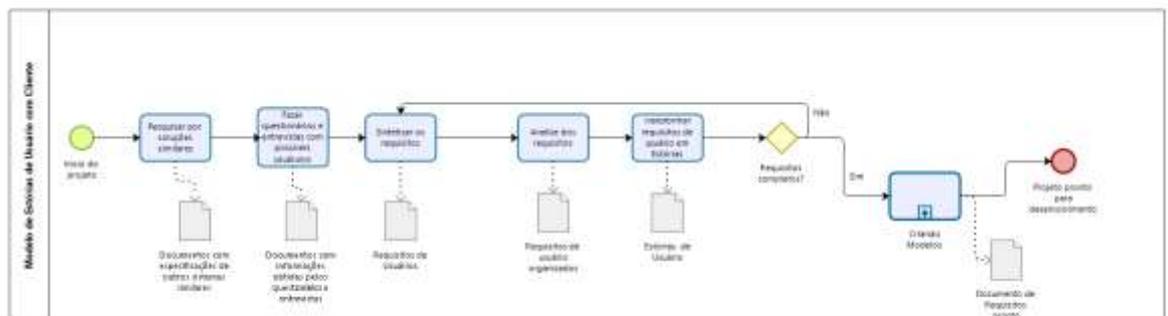
## 2. Objetivo

O processo tem como objetivo dar aos usuários do guia um processo ágil e bem simplificado, mas que ainda assim possua os quesitos de qualidade proposto pelo CMMI e MPS Br. Assim fornecendo agilidade, facilidade e qualidade.

## 3. Modelo Gráfico

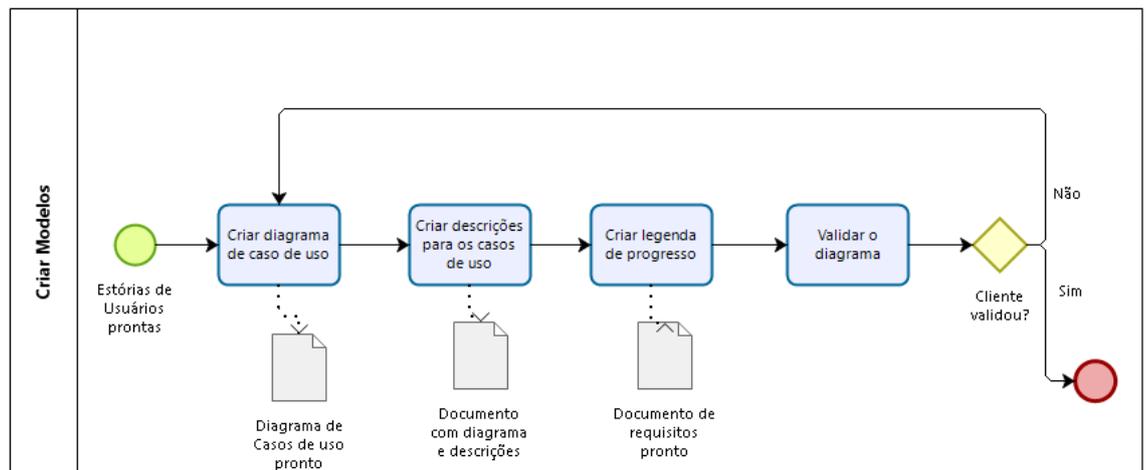
Nesta sessão iremos ver o modelo gráfico do processo de transformação de Estórias de Usuário em diagramas, no modelo que é mostrado na Figura 1 e na Figura 2 será utilizado o diagrama de casos de uso.

Figura 1 - Processo de Estórias de Usuários.



Fonte: Autor do trabalho.

Figura 2 - Subprocesso de Estórias de Usuário.



Fonte: Autor do trabalho.

#### 4. Descrição das Atividades

Nesta sessão veremos a descrição das atividades, eventos e artefatos mostrados nos processos mostrados na Figura 1 e na Figura 2, será detalhado na mesma ordem de apresentação do modelo, mostrando em uma lista, além de incluir ferramentas que podem ser úteis no processo. Cada atividade, artefato ou evento mostrado anteriormente será listado e estarão destacados no texto em negrito.

- **Início do projeto:** O evento de start do processo, ocorre quando o desenvolvedor encontra uma oportunidade.
- **Pesquisar por soluções similares:** A atividade é realizada para que o desenvolvedor encontre softwares existentes que sejam semelhantes, essa pesquisa deve dar noções do que já existe e do que pode ser feito de novo em seu sistema.
- **Documento com especificações de outros sistemas similares:** Documento que resulta da pesquisa feita sobre os softwares existentes, ele contém as informações mais importantes sobre as funcionalidades que já são utilizadas, o documento pode já apresentar ideias com funcionalidades novas.
- **Fazer questionários e entrevistas com possíveis usuários:** Com os resultados do documento anterior, deve ser elaboradas entrevistas e questionários para serem aplicados em possíveis usuários, para identificar o que eles acham do que já existe e de como reagem a mudanças, incluindo o que eles mesmo querem que mude, lembrando evite pedir que escrevam, procure sempre usar perguntas que eles possam marcar.
- **Documentos com informações obtidas pelos questionários e entrevistas:** Documento que reúne as respostas dos possíveis usuários e as divide em porcentagem.
- **Sintetizar os requisitos:** A atividade consiste em pegar todas as informações coletadas anteriormente e transformá-las em requisitos de usuário, para que a partir delas as histórias possam ser feitas.

- **Requisitos de Usuários:** Documento que contém os resultados feito pela atividade de sintetizar os requisitos, requisitos informais e desorganizados, porém são a base das histórias de usuário, qualquer requisito que pode ser mal interpretado futuramente deve ser esclarecido o quanto antes com os clientes e usuários.
- **Análise dos Requisitos:** Nesta atividade a equipe deve documentar os requisitos de usuário, de forma a remover os ambíguos, as repetições e os ilegíveis, porém ainda será mantido em formato de requisitos de usuário, apenas será feito revisões ortográficas e deixando-os mais legíveis e já colocando níveis de prioridade.
- **Requisitos de Usuários Organizados:** No final da tarefa os requisitos de usuários estarão todos em um documento, agora com as prioridades pré estabelecidas e organizados em ordem de preferência dos desenvolvedores, recomendado que sejam feitos na ordem de funcionalidades parecidas ou em dependência de funcionalidade, onde a funcionalidade que depende de outra funcionalidade vem logo após a sua dependência.
- **Transformar requisitos de usuários em histórias de usuários:** Neste ponto os desenvolvedores devem usar os requisitos coletados para fazer as histórias de usuário, onde serão separados cada funcionalidade em um cartão, onde no cartão conterà o identificador da história, o nome da história, os atores que vão realizar aquela funcionalidade, a descrição da interação dos atores com a funcionalidade, os requisitos envolvidos, as condições para aquela história ocorrer e a prioridade.
- **Estórias de Usuário:** Um documento com os cartões de histórias de usuário, esse mesmo documento deve conter um sumário, um glossário para os termos e abreviações, uma introdução, pois ele se tornará posteriormente o documento de requisitos.
- **Requisitos estão prontos:** Após finalizar a atividade, os requisitos e histórias de usuários devem ser validadas junto ao cliente ou usuários, caso eles estejam de acordo o processo deve continuar, caso não atualize os documentos com base nos apontamentos sugeridos pelos usuários ou cliente, caso ache necessário refaça a atividade de coletas de requisitos, caso encontre algum requisito que foi deixado de fora da primeira vez, lembre-se que os requisitos podem mudar constantemente.
- **Criando Modelos:** Quando os requisitos estiverem prontos e validados, os desenvolvedores devem começar a fase de criação dos modelos gráficos do sistema.
  - **Estórias de usuários prontas:** Esse evento marca o início da fase de criação do modelo gráfico, assim que as histórias estiverem prontas e validadas o processo está apto a entrar na fase de construção dos diagramas, será focado no de caso de uso, porém se a equipe achar necessário eles podem utilizar os demais diagramas disponíveis, como por exemplo o de classe.
  - **Criar diagramas de caso de uso:** Com base nas histórias de usuários os desenvolvedores devem criar o diagrama de casos de uso, onde deve ser mostrado os atores (Usuários do sistema) interagindo com as funcionalidades, lembre-se de manter o diagrama simples, quanto mais poluído ele estiver mais difícil será para o cliente entender.
  - **Diagrama de casos de uso pronto:** Após a tarefa o diagrama gerado é colocado em uma sessão do documento de requisitos destinada especificamente ao diagrama.
  - **Criar descrições para os casos de uso:** Agora para cada caso de uso que está no diagrama, deve ser feito um quadro com o identificador do caso de uso, com o nome do caso de uso, com os atores, com uma descrição, com as histórias de usuário

envolvidas, com a pré-condição para realizar a atividade, com o resultado que deve ter sido obtido pelo usuário, com o fluxo, principal, fluxo alternativo e de exceção.

- **Documento com diagrama e descrições:** Documento de requisitos com a sessão de descrição dos casos de uso.
- **Criar legendas de progresso:** Nesta atividade uma legenda deve ser adicionada para manter o controle do que já foi desenvolvido, o que está sendo desenvolvido e o que ainda deve ser desenvolvido, pode ser usado por exemplo um esquema de cores para representar.
- **Documento de requisitos pronto:** Agora os desenvolvedores têm a primeira versão do documento de requisitos pronto, lembre-se de manter um controle de versão para as futuras alterações.
- **O cliente validou:** Caso o cliente aponte alguma alteração, refaça as atividades do subprocesso com as alterações solicitadas, caso ele aprove a equipe terá o documento pronto.
- **Projeto pronto para o desenvolvimento:** Com isso o projeto já está pronto para o desenvolvimento do sistema, porém quando houver mudanças a fase de engenharia de requisitos mostrado nesse processo pode ser refeita.

Para esse processo são recomendados usar o trello, uma ferramenta de gestão de tarefas, assim a equipe saberá o que tem que fazer e o que cada um já tem, e o que estão fazendo, é bem simples de se usar e basta ter uma conta no google para usar.

Outra ferramenta são as disponibilizadas pelo google drive, como o docs, assim todos terão acesso aos documentos mais rapidamente, além de poder fazer apresentações e ter planilhas para o uso da equipe quando necessário.

As entrevistas, questionários e observações ficam a critério dos desenvolvedores, pois eles variam facilmente conforme o sistema solicitado, porém ao final deste documento terá em Apêndice um exemplo de um documento de requisito completo usado por este processo.

## 5. Referências

Jaqueira, Aline de Oliveira Prata. Uso de modelos i\* para enriquecer requisitos em métodos ágeis. 2013. 97 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal do Rio Grande do Norte, Natal, 2013.

Martins, Luiz Eduardo Galvão. Uma metodologia de elicitação de requisitos de software baseada na teoria da atividade. 2001. 170p. Tese (doutorado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação, Campinas, SP. Disponível em: <<http://www.repositorio.unicamp.br/handle/REPOSIP/260233>>. Acesso em: 17 ago. 2020.

MPS.BR - Melhoria de Processo do Software Brasileiro: Guia Geral MPS de Software, SofTex, 2020. Disponível em: <<https://softex.br/mpsbr/guias/#guia-sw>>. Acesso em: 17 ago. 2020.

O QUE É CMMI?. ISDBRASIL. Disponível em: <<http://www.isdbrasil.com.br/o-que-e-cmmi.php>>. Acesso em: 17 ago. 2020.

## 6. Apêndices

**APÊNDICE 1 - Exemplo de documento de requisitos**

(Imagem representando a empresa)

**Documento de Requisitos****Sistema de vendas de ingressos de um cinema**

**INTRODUÇÃO**      **2**

**GLOSSÁRIO**      **2**

**REQUISITOS DE USUÁRIO**      **2**

**ESTÓRIAS DE USUÁRIO**      **3**

**DIAGRAMA DE CASOS DE USO**      **9**

**DESCRIÇÕES DE CASOS DE USO**      **9**

Versão	Data	Autor	Descrição
--------	------	-------	-----------

1.0	17/11/2020	Lucas Rafael	Criação do documento, adição da introdução, do glossário, dos requisitos de usuário, das estórias de usuário, do diagrama de casos de uso e das descrições de caso de uso.
-----	------------	--------------	--

## 1. INTRODUÇÃO

O sistema a ser desenvolvido compreende a venda de ingressos de um cinema, de forma a ajudar os funcionários e tornar todo o processo mais rápido e fácil, além de contribuir a manter o controle das vendas, e este documento é dividido em introdução, glossário, onde serão apresentados os termos, convenções, abreviaturas e os identificadores, em seguida será os requisitos de usuário, logo em seguida as estórias de usuário, o diagrama de caso de uso e a descrição de caso de uso.

## 2. GLOSSÁRIO

Nesta seção veremos os termos, abreviações, convenções e identificadores:

Identificadores: Requisitos de Usuário (RS001), Estórias de Usuário(EU001), Casos de Uso(UC001).

Abreviações: ID(Identificador).

## 3. REQUISITOS DE USUÁRIO

Nesta seção veremos os requisitos coletados com o cliente e os usuários, os requisitos serão divididos por funcionalidades e cada um terá sua prioridade.

ID	Descrição	Prioridade
RS001	Quero que o sistema possa ser acessado por mim e os meus funcionários, quero que possamos modificar os dados e cada um pode excluir seus cadastros, mas só eu posso excluir todos.	Essencial
RS002	Quero ter acesso ao balanceamento financeiro do dia e o mensal.	Essencial

RS003	Quero que mostre quantos assentos tem disponível na hora de vender.	Essencial
RS004	Quero que os preços possam ser definidos e sejam mostrados na hora da compra.	Essencial
RS005	Quero que gere o papel com as informações do ingresso, como hora da sessão, preço do ingresso, assento e sala.	Essencial
RS006	Quero ter acesso aos históricos de finanças e relatórios diários e mensais.	Essencial
RS007	Quero que seja possível criar novas sessões de filmes, além de excluir e modificá-la.	Essencial

#### 4. ESTÓRIAS DE USUÁRIO

<b>ID: EU001</b>
<b>Título: “Cadastro no sistema”</b>
<b>Ator:</b> Gerente, funcionário.
Como Gerente/Funcionário quero ser capaz de fazer meu cadastro no sistema. <b>Prioridade:</b> Essencial

**Critérios de Aceitação:**

- 1 - O sistema deve apresentar uma tela para o usuário poder realizar seu cadastro e ser capaz de inserir o nome, e-mail, senha e cpf.
- 2 - Quando o usuário preencher todos os dados, o sistema deve verificar se o cpf é válido.
- 3 - Quando selecionado a opção “Cadastrar” o sistema deve cadastrar os dados do usuário no sistema.

**Requisitos: RS001****ID: EU002****Título: “login no sistema”****Ator:** Gerente, funcionário.

Como Gerente/Funcionário quero ser capaz de fazer login e acessar as funcionalidades do sistema.

**Prioridade:** Essencial**Critérios de Aceitação:**

- 1 - O sistema deve apresentar uma tela para o usuário inserir seus dados de login como e-mail e senha.
- 2 - Quando o usuário preencher os campos de e-mail e senha e clicar no botão de entrar o

sistema deve verificar se existe no banco de dados.

3 - Caso exista no banco de dados o sistema efetua o login, caso contrário avisa que o usuário não possui cadastro.

**Requisitos: RS001**

**ID: EU003**

**Título: “Modificar Cadastro”**

**Ator:** Gerente, funcionário.

Como Gerente/Funcionário quero ser capaz de fazer alterações no meu cadastro no sistema.

**Prioridade:** Essencial

**Crterios de Aceitaço:**

1 - Depois de já estar logado no sistema o usuário pode clicar no ícone de perfil e lá deve ser capaz de fazer alterações nos seus dados cadastrais.

2 - Quando o usuário finalizar as alterações o sistema deve verificar se as alterações são válidas.

3 - Quando selecionado a opção “Salvar” o sistema deve alterar a as informações no banco de dados.

**Requisitos: RS001**

**ID: EU004**

**Título: “Excluir cadastro no sistema”**

**Ator:** Gerente, funcionário.

Como Gerente/Funcionário quero ser capaz de excluir meu cadastro no sistema.

**Prioridade:** Essencial

**Crterios de Aceitaço:**

1 - Com o usuário logado no sistema, deva ser capaz de excluir o seu próprio perfil, somente o gerente deve ser capaz de excluir os perfis dos funcionários.

**Requisitos: RS001**

**ID: EU005**

**Título: “Acessar Relatários”**

**Ator:** Gerente.

Como Gerente quero ser capaz de acessar relatórios mensais e diários informando os balanços financeiros das vendas.

**Prioridade:** Essencial

**Critérios de Aceitação:**

- 1 - O sistema deve apresentar uma tela para o usuário acessar os relatórios gerados.
- 2 - Quando o usuário acessar a página de relatórios o sistema deve gerar um PDF mostrando o histórico de vendas do dia.
- 3 - Quando o usuário selecionar a opção “Relatório Mensal” o sistema deve gerar um PDF com o histórico financeiro do mês em questão.

**Requisitos:** RS002

**ID:** EU006

**Título:** “Verificar disponibilidade”

**Ator:** Funcionário.

Como Funcionário quero ser capaz de ver quantos assentos ainda tem disponível para determinado filme e sessão.

<b>Prioridade:</b> Essencial
<p><b>Cr�terios de Aceita�o:</b></p> <p>1 - Quando o Usu�rio acessar a parte de venda de um determinado filme e sess�o o sistema deve mostrar quantos acentos ainda tem dispon�vel.</p>
<b>Requisitos:</b> RS003

<b>ID:</b> EU007
<p><b>T�tulo:</b> “Vender Ingressos”</p> <p><b>Ator:</b> Funcion�rio.</p> <p>Como Funcion�rio quero ser capaz de definir os pre�os dos ingressos e quando for para vender eu possa escolher entre os pre�os de um ingresso inteiro ou um ingresso com desconto. <b>Prioridade:</b> Essencial</p>
<p><b>Cr�terios de Aceita�o:</b></p> <p>1 - O sistema deve permitir que o usu�rio seja capaz de definir os pre�os toda vez que uma nova sess�o seja criada.</p> <p>2 - O sistema deve apresentar os pre�os definidos pelo usu�rio quando ele for vender um ingresso.</p>

**Requisitos: RS004**

**ID: EU008**

**Título: “Gerar Ingresso”**

**Ator:** Funcionário.

Como Funcionário quero que o sistema seja capaz de gerar um ingresso quando eu confirmar a venda.

**Prioridade:** Essencial

**Crterios de Aceitaço:**

1 - O sistema deve ser capaz de gerar um ingresso com as informaes de filmes, preo e sesso aps o usurio confirmar a venda.

**Requisitos: RS005**

**ID: EU009**

**Título: “Visualizar Histrico”**

**Ator:** Gerente.

Como Gerente quero ser capaz de visualizar o histórico financeiro dos meses anteriores.

**Prioridade:** Essencial

**Critérios de Aceitação:**

1 - O sistema deve apresentar a opção de gerar um histórico dos meses anteriores.

**Requisitos:** RS006

**ID:** EU010

**Título:** “Cadastrar sessão”

**Ator:** Gerente, funcionário.

Como Gerente/Funcionário quero ser capaz de cadastrar uma nova sessão no sistema.

**Prioridade:** Essencial

**Critérios de Aceitação:**

1 - O sistema deve permitir que os usuários que estejam logados cadastrem novas sessões de filmes, com o título do filme, duração, hora de início da sessão, tempo em cartaz, valor do ingresso e sala da sessão.

2 - Quando o usuário selecionar a opção de “Salvar” o sistema deve verificar se as informações são válidas.

3 - Se as informações forem válidas o sistema deve salvar a nova sessão no banco de dados, caso contrário o sistema deve informar que não foi possível concluir o cadastro por informações inválidas.

**Requisitos: RS007**

**ID: EU011**

**Título: “Modificar Sessão”**

**Ator:** Gerente, funcionário.

Como Gerente/Funcionário quero ser capaz de modificar as sessões existentes no sistema. **Prioridade:** Essencial

**Critérios de Aceitação:**

1 - Com o usuário logado no sistema, deve ser capaz de modificar os dados de uma determinada sessão.

2 - Quando o usuário selecionar a opção de “Salvar” as modificações, o sistema deve verificar se as modificações são válidas.

3 - Se as modificações forem válidas o sistema deve salvar as modificações no banco de

dados, caso contrário deve informar ao usuário que existem informações inválidas.

**Requisitos: RS007**

**ID: EU012**

**Título: “Excluir Sessão”**

**Ator:** Gerente, funcionário.

Como Gerente/Funcionário quero ser capaz de excluir uma sessão existente do sistema.

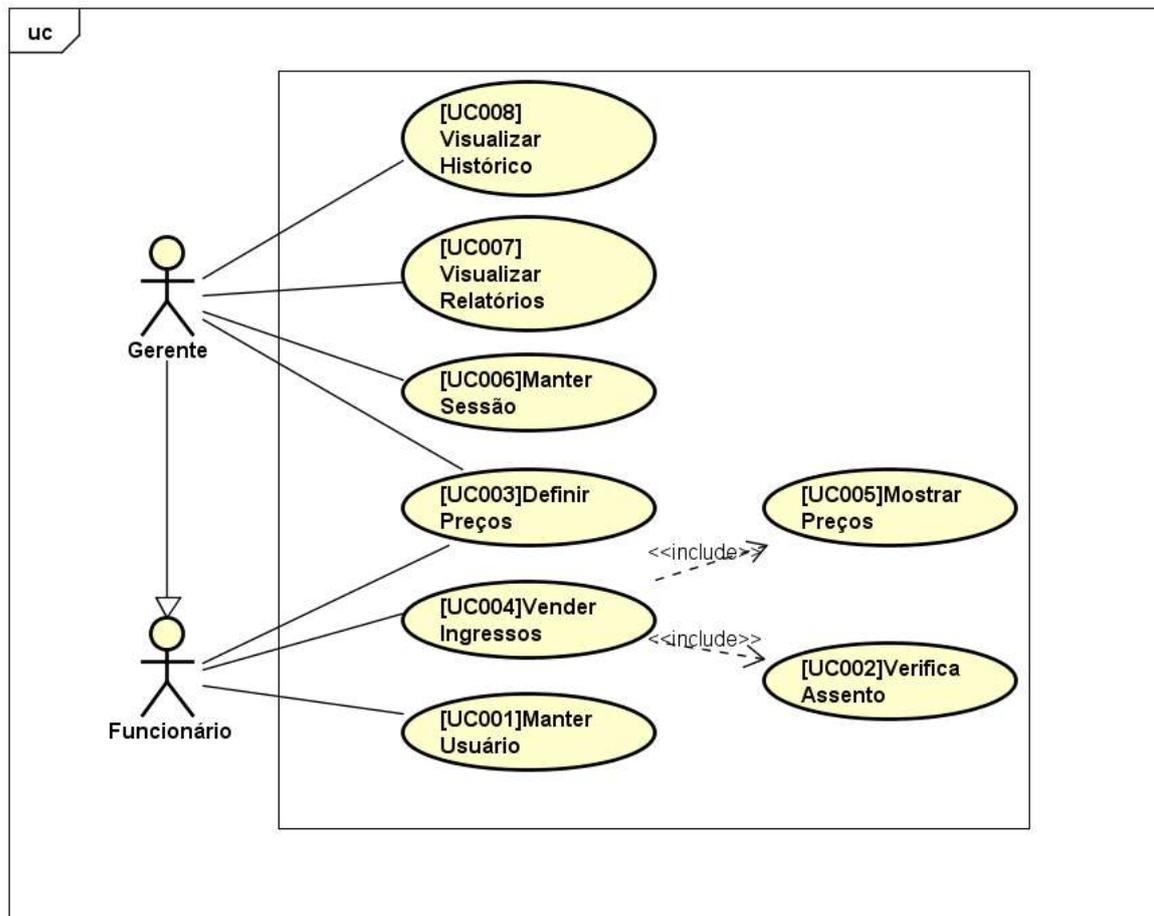
**Prioridade:** Essencial

**Critérios de Aceitação:**

- 1 - O usuário deve estar logado no sistema para conseguir excluir uma sessão.
- 2 - O usuário deve selecionar a opção “Excluir” e o sistema deve remover a sessão do banco de dados.

**Requisitos: RS007**

## 5. DIAGRAMA DE CASOS DE USO



powered by Astah

## 6. DESCRIÇÕES DE CASOS DE USO

Essa seção destina-se a realizar a descrição detalhada dos casos de uso identificados no sistema de venda do sistema. As caixas de texto são divididas em cinco categorias que dizem respeito ao andamento do desenvolvimento dos componentes de venda do cinema. O nível de desenvolvimento do componente é exibido na legenda abaixo:



**Componente desenvolvido**



**Componente em desenvolvimento**



**Componente desenvolvido, mas apresentou falhas**



**Componente ainda não desenvolvido**



**Situação não identificada**

Lista de casos de uso descritos:

ID	UC001
Nome	Manter Usuário
Ator Primário	Gerente, Funcionário
Sumário	Este caso de uso tem como objetivo mostrar o passo a passo que o usuário deve fazer para se cadastrar, logar, modificar e excluir.
Estória de Usuário	[EU001], [EU002], [EU003] e [EU004]
Pré-condições	O usuário ter, ou fazer um novo cadastro.

Pós-condições	Um perfil novo cadastrado, modificado, excluído ou logado
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O sistema apresenta os ícones de “entrar” ou fazer “novo cadastro”.</li> <li>2. O usuário seleciona a opção de “entrar”.</li> <li>3. O sistema exibe a página de login, com o campo “e-mail”, “senha” e os botões de “Entrar” e “Criar novo”</li> <li>4. O usuário insere os dados e seleciona em “Entrar”.</li> <li>5. O sistema verifica a existência do usuário e carrega a página inicial.</li> <li>6. O caso de uso é encerrado.</li> </ol>
Fluxo Alternativo	<p>A2.1 - O usuário seleciona a opção de “novo cadastro”.</p> <p>A2.2 - O sistema exibe uma página com os campos, “nome”, “e-mail”, “cpf”, “cargo” e os botões de "Criar" e “Entrar”.</p> <p>A2.3 - O usuário preenche os dados e seleciona a opção “Criar”.</p> <p>A2.4 - O sistema cria o usuário no banco de dados.</p> <p>A2.5 - O caso de uso retorna para o fluxo principal no passo 5.</p> <p>A3.1 - O usuário seleciona a opção “Criar novo”.</p> <p>A3.2 - O caso de uso volta para o fluxo alternativo A2.</p> <p>A5.1 - O sistema não encontra o usuário.</p> <p>A5.2 - O sistema notifica que o usuário não foi encontrado.</p> <p>A5.3 - O caso de uso retorna ao fluxo principal no passo 3.</p>

Fluxo de Exceção	EA2.2.1 - O usuário seleciona a opção “Entrar”. EA2.2.2 - O caso de uso retorna ao fluxo principal no passo 3.
------------------	---

ID	UC002
Nome	Verificar Assento
Ator Primário	Realizada ou vender ingresso
Sumário	Este caso de uso tem como objetivo mostrar como deve ser o passo a passo que o sistema deve realizar para verificar os assentos do cinema
Estórias de Usuários	[EU006]
Pré-condições	O usuário deve estar logado e efetuando a venda de um ingresso
Pós-condições	O usuário deve saber quais assentos estão disponíveis
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O usuário acessa o menu de vendas.</li> <li>2. O sistema apresenta a tela pedindo as informações da sessão.</li> <li>3. O usuário preenche as informações.</li> <li>4. O sistema verifica através de uma tabela quais assentos ainda estão disponíveis</li> <li>5. O sistema apresenta uma lista de assentos vagos.</li> </ol>

	<p>6. O usuário seleciona um assento.</p> <p>7. O sistema salva a informação.</p> <p>8. O caso de uso é encerrado.</p>
Fluxo Alternativo	<p>A4.1 - O sistema verifica através de uma tabela quais assentos ainda estão disponíveis.</p> <p>A4.2 - O sistema indica que não há mais assentos vagos.</p> <p>A4.3 - O caso de uso retorna ao fluxo principal no passo 7.</p>
Fluxo de Exceção	

<b>ID</b>	<b>UC003</b>
Nome	Definir Preços
Ator Primário	Vendedor, Gerente
Sumário	Este caso de uso tem como objetivo mostrar o passo a passo que o usuário deve fazer para definir os preços dos ingressos.
Estórias de Usuários	[EU007]
Pré-condições	O usuário está previamente logado no sistema e existe pelo menos uma

	sessão cadastrada.
Pós-condições	Preços definidos
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O usuário seleciona a opção “Sessão”</li> <li>2. O sistema exibe um submenu.</li> <li>3. O usuário seleciona a opção “Definir preço”</li> <li>4. O sistema apresenta os campos de preços para os preços e a opção “definir”.</li> <li>5. O usuário preenche os campos de preço "inteiro" e “meia” e seleciona a opção de “definir”</li> <li>6. O sistema salva as informações de preços para aquela sessão.</li> <li>7. O sistema sugere para o usuário se ele quer definir os mesmos preços para todas as sessões e as opções “sim” e “não”.</li> <li>8. O usuário seleciona “sim”</li> <li>9. O sistema retorna à página inicial.</li> <li>10. O caso de uso é encerrado.</li> </ol>
Fluxo Alternativo	<p>A7.1 - O sistema sugere que os preços sejam os mesmos para todas as sessões e as opções “sim” e “não”.</p> <p>A7.2 - O usuário seleciona a opção “não”.</p> <p>A7.3 - O caso de uso retorna ao fluxo principal no passo 9.</p>
Fluxo de Exceção	

ID	UC004
Nome	Vender Ingresso
Ator Primário	Funcionário, Gerente
Sumário	Este caso de uso tem como objetivo mostrar o passo a passo que o usuário deve fazer para executar a funcionalidade de vender ingressos
Estória de Usuário	[EU007], [EU008]
Pré-condições	Usuário está cadastrado e logado no sistema
Pós-condições	Pelo menos um ingresso ter sido vendido
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O usuário acessa o menu venda.</li> <li>2. O sistema solicita que o usuário indique a sessão do ingresso.</li> <li>3. O usuário indica a sessão.</li> <li>4. O sistema apresenta os preços que o usuário pode selecionar e a opção de definir um novo preço, conforme o caso de uso [UC003] e [UC005].</li> <li>5. O usuário seleciona um dos preços existentes.</li> <li>6. O sistema mostra os assentos disponíveis conforme o caso de [UC002].</li> <li>7. O usuário seleciona o assento escolhido e confirma a venda.</li> <li>8. O sistema gera um ingresso com as informações da sessão,</li> </ol>

	<p>(nome do filme, horário, sala e preço).</p> <p>9. O sistema retorna à página inicial.</p> <p>10. O caso de uso é encerrado.</p>
Fluxo Alternativo	<p>A4.1 - O usuário seleciona a opção de definir um novo preço.</p> <p>A4.2 - O sistema realiza os passos do caso de uso [UC003].</p> <p>A4.3 - O caso de uso retorna ao fluxo principal no passo 5.</p> <p>A7.1 - O sistema não encontra assentos disponíveis e executa os passos conforme o fluxo alternativo do caso de uso [UC003].</p>
Fluxo de Exceção	

<b>ID</b>	<b>UC005</b>
Nome	Mostrar preços
Ator Primário	Realizado ao vender ingresso.
Sumário	Este caso de uso tem como objetivo mostrar o passo a passo que o sistema deve fazer para mostrar os preços dos ingressos da sessão selecionada para que o usuário possa marcar o preço desejado
Estórias de	[UC007].

Usuário	
Pré- condições	O usuário está cadastrado, logado e efetuando uma venda de ingresso.
Pós- condições	Um dos preços escolhidos ou novos preços definidos.
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O usuário está no processo de venda de um ingresso e seleciona a caixa de texto reservada para preço.</li> <li>2. O sistema verifica se existem preços definidos na sessão selecionada e apresenta os dois preços e a opção de definir novos preços.</li> <li>3. O usuário seleciona um dos preços.</li> <li>4. O caso de uso é encerrado.</li> </ol>
Fluxo Alternativo	<p>A2.1 - O sistema não identifica nenhum preço definido na sessão.</p> <p>A2.2 - O sistema solicita que o usuário defina os preços dos ingressos.</p> <p>A2.3 - O usuário seleciona a opção de definir preços e preenche os valores de “Inteira” e “Meia” dos ingressos e seleciona a opção “Confirmar”.</p> <p>A2.4 - O caso de uso retorna ao fluxo principal no passo 3.</p> <p>A3.1 - O usuário seleciona a opção de definir novos preços.</p> <p>A3.2 - O sistema exibe caixas de texto com os identificadores “Inteira” e “Meia” e a opção “Confirmar”.</p> <p>A3.3 - O usuário preenche os campos e seleciona a opção de “Confirmar”.</p> <p>A3.4 - O sistema salva as informações dos novos preços na sessão e pergunta ao usuário se os novos preços devem ser aplicados nas demais</p>

	<p>sessões, junto com as opções “Sim” e “Não”.</p> <p>A3.5 - O usuário seleciona a opção “Sim”.</p> <p>A3.6 - O sistema altera todas as sessões existentes para ter os mesmos valores de ingressos.</p> <p>A3.7 - O caso de uso retorna ao fluxo principal no passo 4.</p>
Fluxo de Exceção	<p>EA3.4.1 - O sistema salva as informações dos novos preços na sessão e pergunta ao usuário se os novos preços devem ser aplicados nas demais sessões, junto com as opções “Sim” e “Não”.</p> <p>EA3.4.2 - O usuário seleciona a opção “Não”.</p> <p>EA3.4.3 - O caso de uso retorna ao fluxo principal no passo 4.</p>

<b>ID</b>	<b>UC006</b>
Nome	Manter Sessão
Ator Primário	Gerente
Sumário	Este caso de uso tem como objetivo mostrar o passo a passo que o usuário deve fazer para cadastrar, modificar e excluir uma sessão.
Estórias de Usuários	[UC010], [UC011] e [UC012].
Pré-condições	O usuário deve estar previamente cadastrado e logado no sistema.

Pós-condições	Uma sessão deve ter sido cadastrada, modificada ou excluída com sucesso.
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O usuário vai até o menu sessão.</li> <li>2. O sistema apresenta um submenu: Cadastrar, Editar e Excluir.</li> <li>3. O usuário seleciona a opção de cadastrar.</li> <li>4. O sistema exibe uma página com um formulário com os campos, Filme, Horário, Sala, Qtdd de Assentos e Duração e a opção de “Salvar”.</li> <li>5. O usuário preenche as informações e seleciona a opção “Salvar”.</li> <li>6. O sistema salva as informações e cria uma tabela de assentos para a sessão.</li> <li>7. O sistema retorna à página inicial.</li> <li>8. O caso de uso é encerrado.</li> </ol>
Fluxo Alternativo	<p>A2.1.1 - O usuário seleciona a opção editar.</p> <p>A2.1.2 - O sistema abre uma página com as sessões existentes e a opção de “Editar” ao lado de cada uma.</p> <p>A2.1.3 - O usuário seleciona uma das sessões e seleciona a opção de “Editar” aquela sessão.</p> <p>A2.1.4 - O sistema exibe o formulário com os dados da sessão, Filme, Horário, Sala, Qtdd de Assentos, Duração e as opções “Salvar” e “Cancelar”.</p> <p>A2.1.5 - O usuário modifica as informações necessárias e seleciona a opção “Salvar”.</p> <p>A2.1.6 - O sistema salva as novas informações da sessão.</p> <p>A2.1.7 - O caso de uso retorna ao fluxo principal no passo 7.</p>

	<p>A2.2.1 - O usuário seleciona a opção excluir.</p> <p>A2.2.2 - O sistema abre uma página com as sessões existentes e a opção de “Excluir” ao lado de cada uma.</p> <p>A2.2.3 - O usuário seleciona uma das sessões e seleciona a opção de “Excluir” aquela sessão.</p> <p>A2.2.4 - O sistema pergunta se o usuário tem certeza se quer excluir a sessão e as opções “Sim” e “Não”.</p> <p>A2.2.5 - O usuário seleciona a opção “Sim”.</p> <p>A2.2.6 - O sistema remove as informações da sessão do banco de dados.</p> <p>A2.2.7 - O caso de uso retorna ao fluxo principal no passo 7.</p>
Fluxo de Exceção	<p>EA2.1.5.1 - O usuário seleciona a opção “Cancelar”.</p> <p>EA2.1.5.2 - O sistema retorna à página com as sessões existentes.</p> <p>EA2.1.5.3 - O caso de uso é encerrado.</p> <p>EA2.2.5.1 - O usuário seleciona a opção “Não”.</p> <p>EA2.2.5.2 - O sistema retorna à página com as sessões existentes.</p> <p>EA2.2.5.3 - O caso de uso é encerrado.</p>

<b>ID</b>	<b>UC007</b>
Nome	Visualizar Relatórios
Ator Primário	Gerente

Sumário	Este caso de uso tem como objetivo mostrar o passo a passo que o usuário deve fazer para visualizar os relatórios de venda do cinema.
Estórias de Usuários	[EU005].
Pré-condições	O usuário deve estar previamente cadastrado e logado no sistema.
Pós-condições	Ter pelo menos um relatório gerado.
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O usuário seleciona o menu relatório.</li> <li>2. O sistema gera uma página em PDF com o relatório do dia, contendo as informações de vendas e a opção de “Baixar Relatório” e “Gerar Relatório Mensal”.</li> <li>3. O usuário seleciona a opção de “Baixar relatório”.</li> <li>4. O sistema gera um arquivo PDF com o relatório do dia.</li> <li>5. O caso de uso é encerrado.</li> </ol>
Fluxo Alternativo	<p>A3.1 - O usuário seleciona a opção de “Gerar Relatório Mensal”.</p> <p>A3.2 - O sistema exibe uma página em PDF com o relatório de vendas de todos os dias daquele mês em questão e a soma do valor bruto total de venda do mês e a opção de “Baixar PDF”.</p> <p>A3.3 - O usuário seleciona a opção de “Baixar PDF”.</p> <p>A3.4 - O sistema gera um arquivo PDF com o relatório do mês.</p> <p>A3.5 - O caso de uso é encerrado.</p>
Fluxo de	

Exceção	
---------	--

<b>ID</b>	<b>UC008</b>
Nome	Visualizar Histórico.
Ator Primário	Gerente.
Sumário	Este caso de uso tem como objetivo mostrar o passo a passo que o usuário deve fazer para visualizar os históricos de vendas.
Estórias de Usuários	[EU009]
Pré-condições	O usuário deve estar previamente cadastrado e logado no sistema.
Pós-condições	O usuário deve ter conseguido ter acesso ao histórico de vendas.
Fluxo Principal	<ol style="list-style-type: none"> <li>1. O usuário seleciona o menu de Histórico.</li> <li>2. O sistema apresenta um calendário para que o usuário selecione o período de início e de fim, para que o sistema mostre as informações do período.</li> <li>3. O usuário seleciona o período.</li> <li>4. O sistema exibe um PDF com as informações de venda dos dias do período e do total de cada mês do período e as opções “Baixar PDF” e “Verificar outro período”.</li> <li>5. O usuário seleciona a opção de “Baixar PDF”.</li> </ol>

	<p>6. O sistema gera um arquivo em PDF com o histórico do período.</p> <p>7. O caso de uso é encerrado.</p>
Fluxo Alternativo	<p>A5.1 - O usuário seleciona a opção de “Verificar outro período”.</p> <p>A5.2 - O caso de uso retorna ao fluxo principal no passo 2.</p>
Fluxo de Exceção	

**Apêndice 3 – Modelo de Processo Ágil com Cliente.**

# Modelo de Processo Ágil com Clientes

Modelo proposto por: Lucas Rafael Alves Oliveira

Março de 2021

**SUMÁRIO**

**Introdução**    **3**

**Objetivo**    **3**

**Modelo Gráfico**    **3**

**Descrição das Atividades**    **4**

**Referências**    **6**

**Apêndices**    **7**

### 1. Introdução

Pensando em empresas que precisam de velocidade em seus processos, o modelo de processo ágil aqui proposto, foi pensado para equipes com pouco experiência e com grande frequência de mudança na equipe, o processo leva em consideração os níveis de qualidade do CMMI e MPS Br, porém o MPS Br será mais amplamente abordado durante o processo.

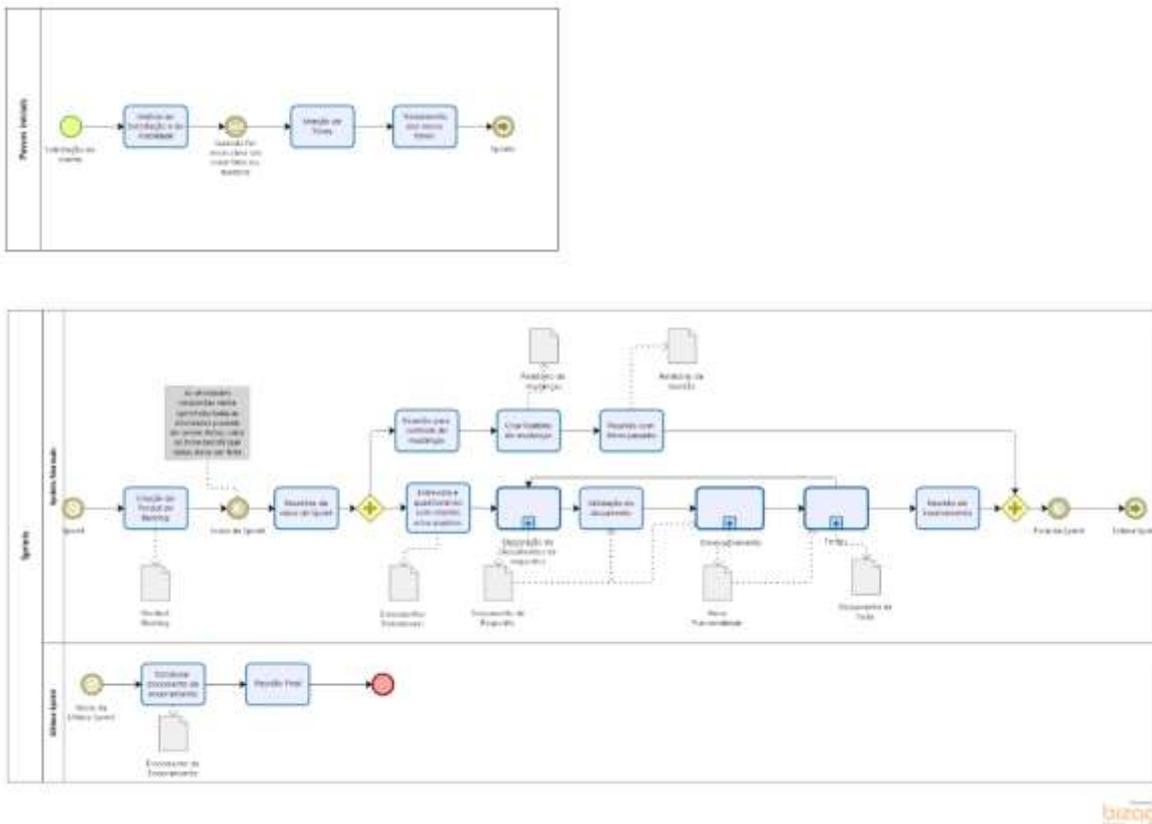
### 2. Objetivo

O processo tem como objetivo dar aos usuários do guia um processo ágil e bem simplificado, mas que ainda assim possua os quesitos de qualidade proposto pelo CMMI e MPS Br. Assim fornecendo agilidade, facilidade e qualidade.

### 3. Modelo Gráfico

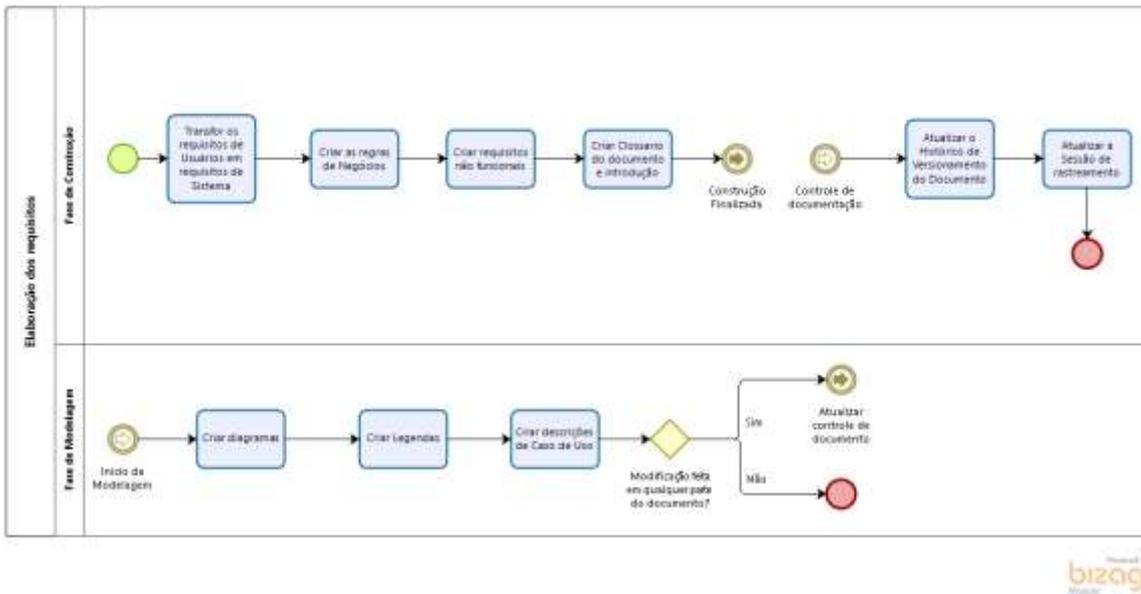
Nesta sessão iremos ver o modelo gráfico do processo ágil com cliente, focando na frequência de novos membros na equipe, como é mostrado nas Figuras 1, 2, 3 e 4.

Figura 1 - Processo de Ágil.



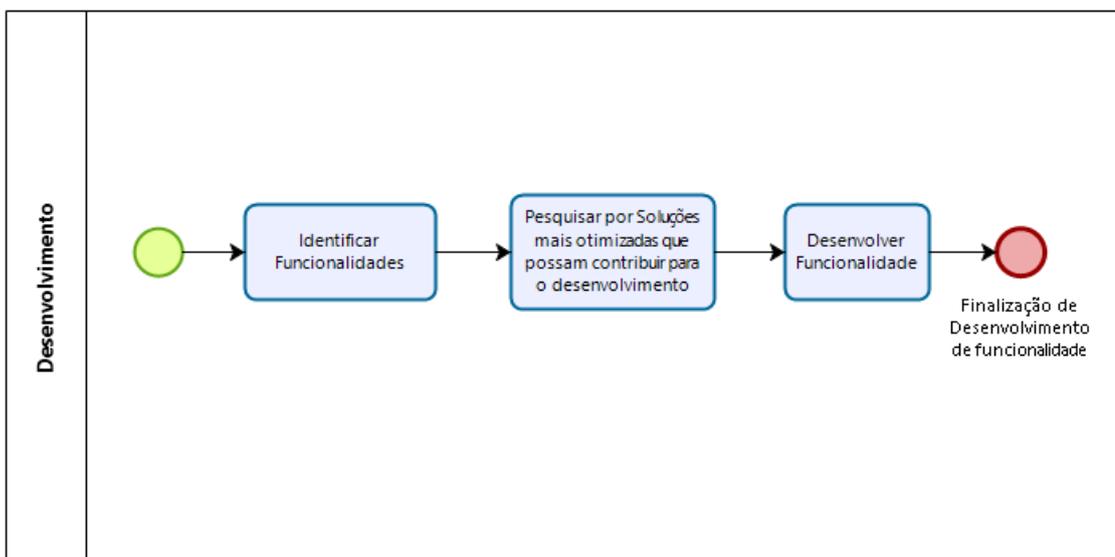
Fonte: Autor do trabalho.

Figura 2 - Modelo Ágil com Cliente Subprocesso Requisitos.



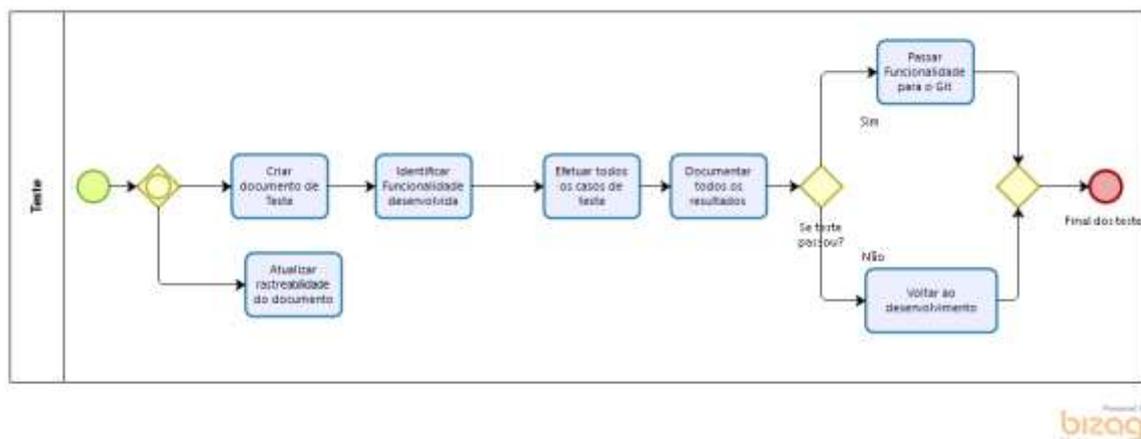
Fonte: Autor do trabalho.

Figura 3 - Modelo Ágil com Cliente Subprocesso Desenvolvimento.



Fonte: Autor do Trabalho.

Figura 4 - Modelo Ágil com Cliente Subprocesso Teste.



Fonte: Autor do Trabalho.

#### 4. Descrição das Atividades

Nesta sessão veremos a descrição das atividades, eventos e artefatos mostrados nos processos mostrados nas Figuras 1, 2, 3 e 4, será detalhado na mesma ordem de apresentação do modelo, mostrado em uma lista, além de incluir ferramentas que podem ser úteis no processo. Cada atividade, artefato ou evento mostrado anteriormente será listado e estarão destacados no texto em negrito.

- **Solicitação do cliente:** O evento de start do processo, ocorre quando o cliente envia uma solicitação para a equipe.
- **Análise de Solicitação e Viabilidade:** Os responsáveis fazem uma análise da solicitação e a viabilidade para saber se o sistema solicitado é possível de ser feito, caso seja então a solicitação passa para uma equipe que estará responsável por desenvolver o sistema.
- **Quando for necessário um novo time ou membro:** O evento pode ocorrer a cada início de ano, 6 meses, ou até mesmo com maior frequência, sempre que acontecer o evento começa e a parte do processo a seguir é executada.
- **Seleção de Times:** Quando for necessário que novos times ou membros sejam selecionados, então a empresa abre um processo seletivo da melhor forma possível para a empresa.
- **Treinamento dos novos times:** Os novos membros devem passar por um treinamento, incluindo as tecnologias que a empresa usa, como o trello, a tecnologias de desenvolvimento, Git, sobre o processo, os documentos, e uma um tempo para conhecer o sistema.
- **Sprint:** Evento que ocorre logo após os treinamentos, os membros começaram a participar efetivamente das sprints.
- **Criação do Product Backlog:** Os membros do time irão se reunir para decidir quais são as principais atividades que serão feitas nas Sprints e criar um documento, ou usar um quadro do trello para comportar todas as atividades que serão realizadas.
- **Início da Sprint:** Evento que marca o início das Sprints enquanto os times estão alocados aos projetos, as atividades após o evento podem ser refeitas em qualquer sprint e podem ser feitas em paralelas com outras atividades.

- **Reunião do Início da Sprint:** Em todo início da Sprint deve haver uma reunião entre os membros da equipe com os responsáveis, para decidir quais serão as atividades que serão feitas durante a Sprint, incluindo fazer a estimativa de tempo que cada atividade levará para ser concluída durante a Sprint, além de se que é recomendado fazer as atividades com maior prioridade primeiro.
- **Entrevista e questionários com cliente e/ou usuários:** Nesta atividade membros das equipes devem ver como é o processo que os clientes ou usuários fazem sem o sistema e também fazer entrevistas e questionários sobre as funcionalidades que os clientes e usuários desejam que seja implementada.
- **Documentos Preliminares:** Um documento com as informações das entrevistas e as respostas dos questionários, essas informações vão conter, os requisitos funcionais, requisitos não funcionais, regras de negócio e até as histórias de usuário.
- **Elaboração do documento de requisitos:** Todas as informações do sistema estão neste documento, os requisitos, regras de negócio, diagramas, descrições e se a equipe achar necessário até mesmo histórias de usuário.
  - **Transformar requisitos de usuário, em requisitos de sistema:** Os dados colhidos nas entrevistas e questionários agora passarão a ser os requisitos do sistema, os membros da equipe responsáveis por isso devem tomar bastante cuidado em diferenciar os requisitos funcionais de requisitos não funcionais e regras de negócios, se caso algumas delas se mostrar de difícil entendimento, deve contatar o cliente para tirar as dúvidas, ou se caso for um requisito feito por um membro anterior, caso ele esteja disponível deve entrar em contato com ele, se caso for necessário pode se alterar o requisito.
  - **Criar as regras de negócios:** Com base nos documentos preliminares os responsáveis devem selecionar e identificar e separar as regras de negócio em uma sessão, caso ocorra de ser uma regra de negócio que foi escrita por outra pessoa, tente entrar em contato com o cliente ou com a pessoa que a escreveu.
  - **Criar requisitos não funcionais:** Com os requisitos de sistema, separe o que é requisito funcional e não funcional em sessões diferentes e todos devem conter os identificadores, caso precisem ser mudados, sempre lembrar de atualizar os identificadores, caso tenha dúvidas sobre requisitos existentes entrar em contato com o cliente ou com quem escreveu os requisitos.
  - **Criar Glossário do documento e a introdução:** O documento deve ter uma introdução do que se trata o sistema e como o documento é dividido, com todas as informações disponíveis já é possível criar, além de criar o glossário, uma sessão reservada a explicar os identificadores, abreviações e palavras pouco conhecidas.
  - **Criar diagramas:** Nestas atividades todos os diagramas necessários ao sistema são criados, comumente serão os diagramas de caso de uso, de classe e de estado, mas se a empresa optar por mais pode utilizar quais quiser, todos serão colocados em uma sessão separados por subseções.
  - **Criar legendas:** Já na sessão de descrição de caso de uso, uma legenda será criada para que seja mais facilmente visível as funcionalidades que já foram desenvolvidas ou que não desenvolveu, ou que foram parcialmente desenvolvidas, pode se utilizar cores para diferenciar.
  - **Criar legendas:** Já na sessão de descrição de caso de uso, uma legenda será criada para que seja mais facilmente visível as funcionalidades que já foram desenvolvidas

ou que não desenvolveu, ou que foram parcialmente desenvolvidas, pode se utilizar cores para diferenciar.

- **Criar descrições de caso de uso:** Cada caso de uso deve ter uma descrição correspondente, com todos os detalhes, como, nome, ator, descrição, requisitos e regras de negócio envolvidas, pré e pós condições e os fluxos principais, alternativos e de exceção, além de conter a legenda antes criada.
- **Atualizar o histórico de versionamento:** Sempre que o documento for alterado o histórico deve ser atualizado, com quem atualizou, quando, e quais foram as alterações.
- **Atualizar a Sessão de rastreamento:** O documento deve mostrar todos os documentos que possuem relação com ele, caso tenha, deve ser colocado em uma tabela o identificador do documento e fazer um link externo, o documento de requisitos por exemplo deve ter o identificador DRE junto a numeração corresponde ao documento como por exemplo 001 e assim por diante. Cada documento vai ter um identificador próprio e a numeração não deve ter relação com nenhum outro identificador.
- **Documento de requisitos:** Documento que contém os requisitos funcionais, não funcionais, regras de negócios, estórias de usuário, diagramas e descrições de casos de uso, será utilizado em quase todo o processo.
- **Validação do documento:** Uma reunião entre os membros da equipe é feita para validar o documento, após aos supervisores e ao cliente, caso precise de mudanças, essas devem ser feitas o quanto antes.
- **Desenvolvimento:** Com o documento de requisitos o mais completo possível, devem começar com as atividades de desenvolvimento nas sprints, mas sempre que encontrar algo que precise ser acrescentado ou mudado deve se atualizar o documento nas atividades anteriores.
  - **Identificar Funcionalidades:** Os desenvolvedores devem identificar todas as funcionalidades que precisam ser desenvolvidas, para então decidir entre os desenvolvedores qual ficará com quais funcionalidades durante a sprint.
  - **Pesquisar por soluções mais otimizadas que possam contribuir para o desenvolvimento:** Antes de começar a desenvolver, seria bem recomendado fazer uma pesquisa, para ver o que já existe, talvez algoritmos altamente usados para resolver os problemas, ou que sejam problemas parecidos que possam ser aproveitados na aplicação para agilizar o desenvolvimento.
  - **Desenvolver Funcionalidade:** Com a pesquisa feita, então o desenvolvedor responsável deve começar a desenvolver a funcionalidade com base em suas descrições, após a funcionalidade está completa então deve passar para os testes.
- **Nova Funcionalidade:** É o arquivo ou classe com os códigos das novas funcionalidades.
- **Testes:** Os testes são uma parte importante, aqui algum membro da equipe que não necessariamente seja o desenvolvedor, deve realizar uma sequência de testes documentados para determinar se as funcionalidades estão fazendo o que foi especificado na documentação.
  - **Atualizar rastreabilidade do documento:** A seção do documento que faz referência aos documentos que o documento de teste faz referência, lembrando de diferenciar os identificadores por os respectivos documentos, como por exemplo documento de requisitos com o identificador DRE ou documento de teste DTE, e colocando a

numeração correspondente ao documento e após deve linkar os documentos através de um link externo.

- **Criar documentos de testes:** São os documentos mostra o passo a passo que o testador deve fazer para identificar se o sistema está fazendo o que deve fazer em todos os casos possíveis, além de documentar quem fez o teste, o resultado e caso tenha, uma solução ou sugestão, ou se a funcionalidade realmente faz o que deveria de acordo com as especificações.
  - **Identificar as funcionalidades desenvolvidas:** O testador deve verificar quais foram as funcionalidades desenvolvidas o quanto antes após seu desenvolvimento e fazer os testes.
  - **Efetuar todos os casos de teste:** Os casos de testes foram desenvolvidos com base no documento de requisitos, o testador deve realizar todos os casos para determinar se a funcionalidade faz ou não o que deve.
  - **Documentar todos os resultados:** Com os casos de teste feitos, o testador deve documentar todos os resultados encontrados no documento de teste.
  - **Passar funcionalidade para o Git:** Se caso a funcionalidade estiver funcionando corretamente ela deve ser colocada no repositório da empresa, para que todos os membros da equipe possam ter acesso e ter o controle sobre o desenvolvimento e marcar em todas as legendas das documentações, como completa, ou parcialmente desenvolvida, caso só uma parte dela seja desenvolvida.
  - **Voltar ao desenvolvimento:** O testador deve retornar a funcionalidade para lista de desenvolvimento, e marcar em todas as legendas nas documentações, que aquela funcionalidade ou falhou ou está incompleta.
- **Documento de teste:** Documento onde terá todos os casos de teste e os resultados dos testes realizados em cada funcionalidade, além de conter os nomes de quem realizou os testes e os resultados.
  - **Reunião para controle de mudanças:** Sempre que alguma mudança deve ser feita uma reunião com a equipe e se possível com alguém que seja responsável pelo sistema, para discutir sobre as mudanças necessárias e saber os impactos que causará no sistema e se é viável ou não.
  - **Criar relatório de mudanças:** Um membro da equipe deve criar um documento para identificar quem fez a mudança, por que fez a mudança e quando, sempre que algo for mudado deve ser adicionado a um relatório.
  - **Relatório de mudanças:** Documento com todas as mudanças detalhadas, mostrando o que foi mudado, por quem, justificativa e quando a mudança foi feita, o documento sempre deve se manter atualizado, pois ele deve ser estudado por novos membros das equipes quando entrarem, para entender o motivo de determinadas escolhas de membros anteriores.
  - **Reunião com times passados:** Quando os dados que estão disponíveis não forem o suficiente e membros anteriores estiverem disponíveis para se reunir, sempre é importante ir até eles para tirar dúvidas que persistem na documentação ou no código.
  - **Relatório de reunião:** Para que as informações não se percam, um relatório deve ser escrito, contendo as dúvidas e as respostas obtidas.

- **Reunião de encerramento:** Uma reunião geral, para mostrar o que foi feito durante a sprint, o que foi aprendido e o que pode ser melhorado para a próxima sprint.
- **Estruturar documento de encerramento:** Elaborar documento que contenha as principais informações de como o sistema está ao final da permanência da equipe, e o que precisa ser feito, detalhadamente para que os próximos times agilizem o processo de desenvolvimento.
- **Documento de Encerramento:** Documento com informações sobre como foi que o sistema ficou ao término das Sprints, o que falta fazer, o que ficou incompleto, o que é mais importante, onde podem achar as informações necessárias.
- **Reunião Final:** Reunião de encerramento geral, onde serão apresentados os resultados e o que ficou por fazer.

Para esse processo é recomendado usar o trello, uma ferramenta de gestão de tarefas, assim a equipe saberá o que tem que fazer e o que cada um já faz, e o que estão fazendo, é bem simples de se usar e basta ter uma conta no google para usar.

Outra ferramenta são as disponibilizadas pelo google drive, como o docs, assim todos terão acesso aos documentos mais rapidamente, além de poder fazer apresentações e ter planilhas para o uso da equipe quando necessário.

Por fim o uso do Git para fazer a gestão do sistema, assim mantendo o controle da versão e que todos os membros da equipe tenham acesso aos mesmos códigos.

As entrevistas, questionários e observações ficam a critério dos desenvolvedores, pois eles variam facilmente conforme o sistema solicitado, porém ao final deste documento terá em Apêndice um exemplo de um documento de requisito completo usado por este processo, além do documento de teste e um modelo de relatório de mudanças.

## 5. Referências

CAVALCANTE, Mirlane Beserra. Proposta de um modelo baseado na combinação de metodologias ágeis de desenvolvimento de software. 2018.115 f. Monografia (Graduação em Engenharia de Software) - Campus de Russas, Universidade Federal do Ceará, Russas, 2018.

MARTINS, Luiz Eduardo Galvão. Uma metodologia de elicitação de requisitos de software baseada na teoria da atividade. 2001. 170p. Tese (doutorado) - Universidade Estadual de Campinas, Faculdade de Engenharia Eletrica e de Computação, Campinas, SP. Disponível em: <<http://www.repositorio.unicamp.br/handle/REPOSIP/260233>>. Acesso em: 17 ago. 2020.

MPS.BR - Melhoria de Processo do Software Brasileiro: Guia Geral MPS de Software, SofTex, 2020. Disponível em: <<https://softex.br/mpsbr/guias/#guia-sw>>. Acesso em: 17 ago. 2020.

O QUE É CMMI?. ISDBRASIL. Disponível em: <<http://www.isdbrazil.com.br/o-que-e-cmmi.php>>. Acesso em: 17 ago. 2020.

**6.** Apêndices**APÊNDICE 1 - Exemplo de documento de requisitos**

(Imagem representando a empresa)

## Documento de Requisitos

Projeto: [*Nome do Projeto*]

Russas - Ceará

27/03/2021

## Sumário

<b><u>1. INTRODUÇÃO</u></b>	<b><u>5</u></b>
<u>1.1. Visão geral do documento</u>	<u>5</u>
<u>1.2. Identificação dos requisitos</u>	<u>5</u>
<u>1.3. Prioridades dos requisitos</u>	<u>5</u>
<b><u>2. RASTREAMENTO DE DOCUMENTAÇÃO</u></b>	<b><u>5</u></b>
<b><u>3. DESCRIÇÃO GERAL DO SISTEMA</u></b>	<b><u>5</u></b>
<u>3.1. Sistema</u>	<u>5</u>
<u>3.2. Stakeholders</u>	<u>6</u>
<b><u>4. LISTAGEM DOS REQUISITOS</u></b>	<b><u>6</u></b>
<u>4.1 Regras de Negócio</u>	<u>6</u>
<u>4.2. Estórias de Usuários</u>	<u>6</u>
<u>Requisitos Funcionais</u>	<u>7</u>
<u>4.3. Requisitos Não-Funcionais</u>	<u>7</u>
<b><u>5. MODELAGEM DO SISTEMA</u></b>	<b><u>7</u></b>
<b><u>6. DESCRIÇÃO DE CASO DE USO</u></b>	<b><u>7</u></b>

## HISTÓRICO DE ALTERAÇÕES

*[Ao final de cada modificação no documento o histórico de versão deve ser atualizado, a mudança de versão nem sempre é necessária, apenas quando a alteração for grande o suficiente para isso, além disso manter a prática de manter as linhas alternadas]*

Data	Versão	Descrição	Autor	Alterações
------	--------	-----------	-------	------------

--	--	--	--	--

## GLOSSÁRIO

*[Todos os termos pouco conhecidos e que podem ser estranhos, abreviações e nomenclaturas devem ser colocadas na tabela juntamente com sua descrição como mostrado no exemplo a seguir]*

Segue abaixo a tabela de termos e suas respectivas descrições usadas no sistema:

Termo	Descrição
Requisitos funcionais	Características e ações que o sistema deve realizar para atender a necessidade do cliente.
Requisitos funcionais não	Descrevem atributos e restrições que o sistema deve ter.
<i>Stakeholders</i>	Pessoas ou organizações que serão afetadas pelo sistema e que, direta ou indiretamente, têm influência sobre os requisitos.
<i>Templates</i>	Modelos de documentos que indicam o conteúdo necessário. A Engenharia de <i>Software</i> utiliza-os com muita frequência para sugerir padrões.

## ● 1. INTRODUÇÃO

*[Esta seção do documento é destinada a introdução sobre o que se trata o documento de requisitos, essa sessão ainda será dividida em outras três subseções que serão mais específicas]*

### ○ 1.1. Visão geral do documento

*[Nesta subseção deve ser mostrado como o documento está dividido]*

### ○ 1.2. Identificação dos requisitos

*[Nesta subseção deve ser mostrado como será feito a identificação dos requisitos e explica cada uma delas]*

### ○ 1.3. Prioridades dos requisitos

*[Nesta subseção deve ser apresentado qual será a prioridade dos requisitos e qual a definição de cada prioridade]*

## ● 2. RASTREAMENTO DE DOCUMENTAÇÃO

*[Nesta seção deve ser colocado todos os documentos que serviram de referência direta para a criação deste documento, colocando o identificador do documento, junto com um link externo]*

## ● 3. DESCRIÇÃO GERAL DO SISTEMA

*[Esta seção será dividida em duas subseções]*

### ○ 3.1. Sistema

*[Nesta subseção deve ser feita uma descrição completa sobre o sistema que está sendo desenvolvido, informações como para o que serve, funcionalidades de forma geral, entre outras informações]*

### ○ 3.2. Stakeholders

*[Nesta subseção deve ser descrito quem são todos os Stakeholders do sistema]*

## ● 4. LISTAGEM DOS REQUISITOS

*[Esta seção será dividida em três subseções]*

#### ○ 4.1 Regras de Negócio

*[Nesta subseção as regras de negócio devem ser identificadas e descritas de forma legível]*

ID Regra de Negócio	Título	Descrição

#### ○ 4.2. Estórias de Usuários

*[Nesta subseção as estórias de usuário devem ser colocadas em quadros, especificados, identificados, claros e coesos]*

<b>ID:</b>
<p style="text-align: center;"><b>Título: “”</b></p> <p><b>Descrição:</b></p> <p><b>Prioridade:</b></p>
<b>Critérios de Aceitação:</b>

<b>Testes a Realizar:</b>

**Requisitos Funcionais**

*[Nesta subseção os requisitos funcionais devem ser identificados e descritos de forma legível, coesa e coerente]*

Identificador	Título	Prioridade	Descrição
---------------	--------	------------	-----------

--	--	--	--

**4.3. Requisitos Não-Funcionais**

*[Nesta subseção os requisitos não-funcionais devem ser identificados e descritos conforme as necessidades, legíveis, entendíveis e coerentes]*

Identificador	Título	Descrição
---------------	--------	-----------

--	--	--

--	--	--

- **5. MODELAGEM DO SISTEMA**

*[Nesta seção estarão todos os diagramas feitos para modelar o sistema, cada diagrama será uma subseção desta]*

- **6. DESCRIÇÃO DE CASO DE USO**

*[Nesta seção cada caso de uso deve ser descrito exatamente da forma que ele deve se comportar em todos os cenários possíveis]*

O nível de desenvolvimento do componente é exibido na legenda abaixo:



**Componente desenvolvido**



**Componente em desenvolvimento**



**Componente desenvolvido, mas apresentou falhas**



**Componente ainda não desenvolvido**



**Situação não identificada**

Lista de casos de uso descritos:

ID
----

Nome	
Ator Primário	
Sumário	
Pontos de Extensão	
Regras de Negócio	
Estórias de Usuário	
Requisitos Funcionais	
Pré-condições	
Pós-condições	
Fluxo Principal	
Fluxo Alternativo	
Fluxo de Exceção	

## APÊNDICE 2 - Exemplo de documento de Relatório de Mudança

(imagem da empresa)

Relatório de Mudanças (*nome da empresa/sistema*)

Projeto: [*Nome do Projeto*]

Relatório de Mudanças

Nome	Data	Mudança	Motivo	Rastreamento de Documentação
Lucas Rafael	15/01/2021	A regra de negócio [RN013] foi removida.	A regra de negócio [RN008] já engloba a ocasião da regra de negócio [RN013].	[DR001]
Lucas Rafael	15/01/2021	O ator professor foi removido da documentação.	Pois foi se notado que a parte que condizia com o ator professor já não fazia mais parte do que o sistema deveria fazer.	[DR001], [DMU001], [DTE001]
Lucas Rafael	15/01/2021	O requisito funcional [RF007] foi alterado.	O requisito se mostrava ambíguo e repetia o que o requisito funcional [RF003] fazia, porém de forma confusa.	[DR001]

**Apêndice 4 – Modelo de Processo Ágil com cliente implícito.**

# Modelo de Processo Ágil com Cliente Implícito

Modelo proposto por: Lucas Rafael Alves Oliveira

Março de 2021

**SUMÁRIO**

**Introdução**    **3**

**Objetivo**    **3**

**Modelo Gráfico**    **3**

**Descrição das Atividades**    **4**

**Referências**    **6**

**Apêndices**    **7**

### 1. Introdução

Pensando em empresas que precisam de velocidade em seus processos, o modelo de processo ágil aqui proposto, foi pensado para equipes com pouco experiência e com grande frequência de mudança na equipe, o processo leva em consideração os níveis de qualidade do CMMI e MPS Br, porém o MPS Br será mais amplamente abordado durante o processo.

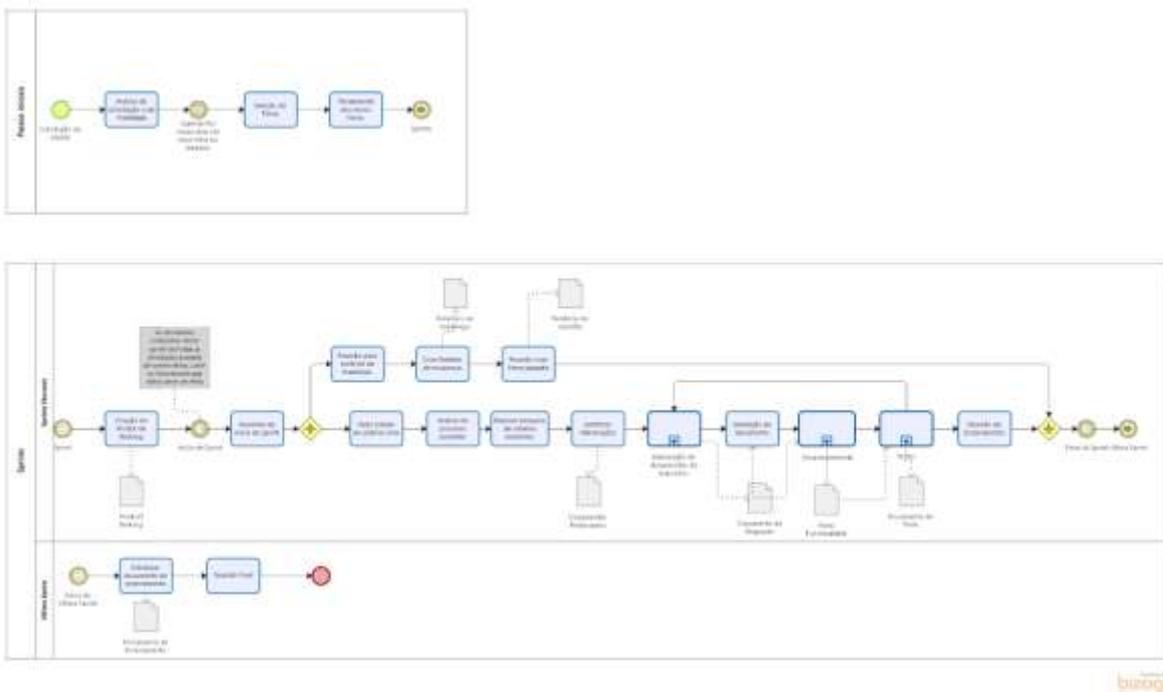
### 2. Objetivo

O processo tem como objetivo dar aos usuários do guia um processo ágil e bem simplificado, mas que ainda assim possua os quesitos de qualidade proposto pelo CMMI e MPS Br. Assim fornecendo agilidade, facilidade e qualidade.

### 3. Modelo Gráfico

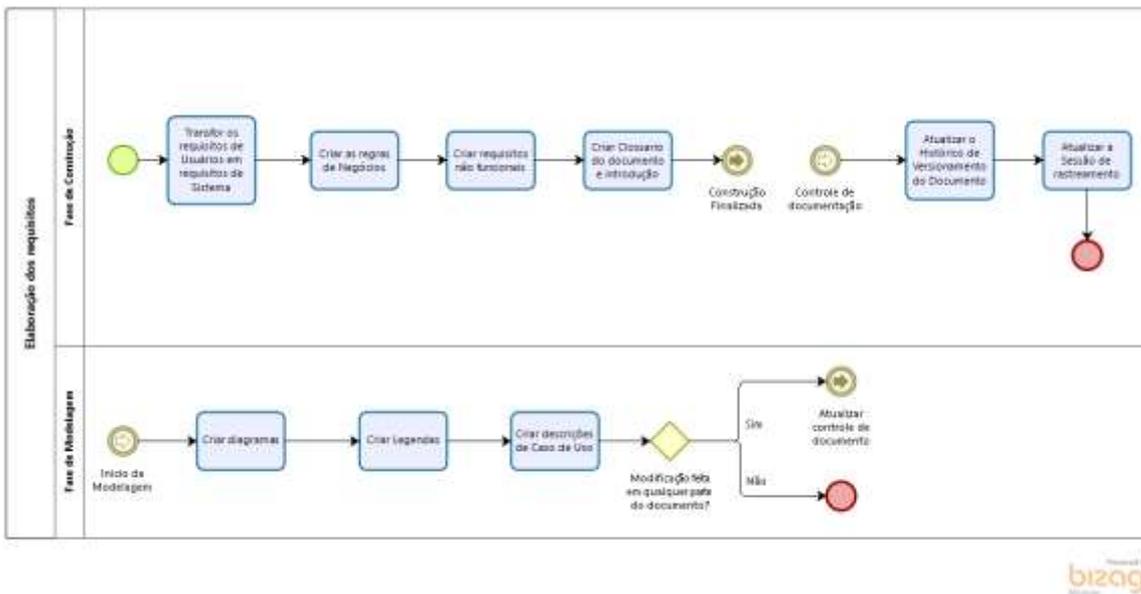
Nesta sessão iremos ver o modelo gráfico do processo ágil com cliente implícito, focando na frequência de novos membros na equipe, como é mostrado nas Figuras 1, 2, 3 e 4.

Figura 1 - Processo de Ágil com cliente implícito.



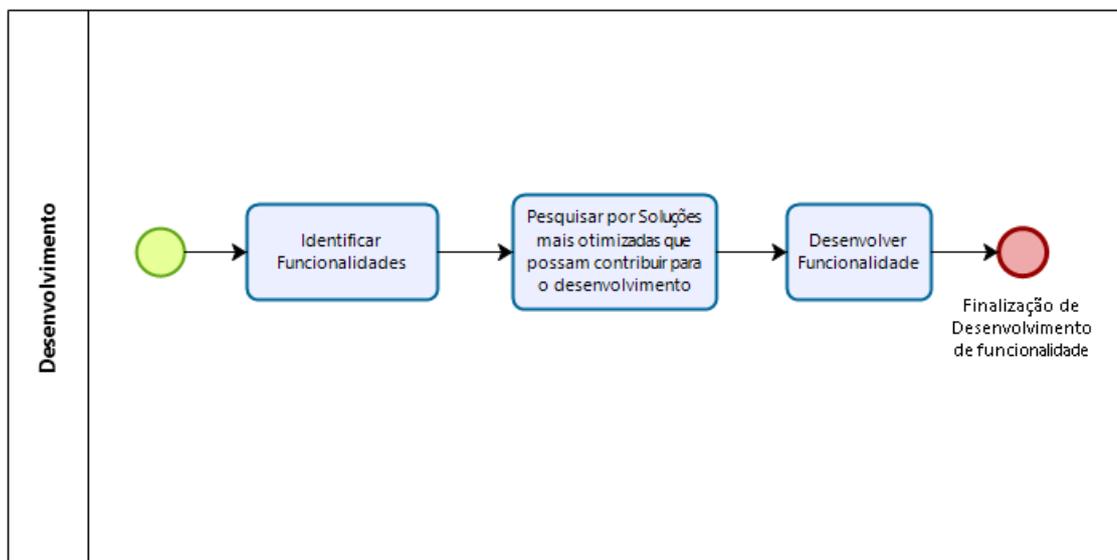
Fonte: Autor do trabalho.

Figura 2 - Modelo Ágil com cliente implícito Subprocesso Requisitos.



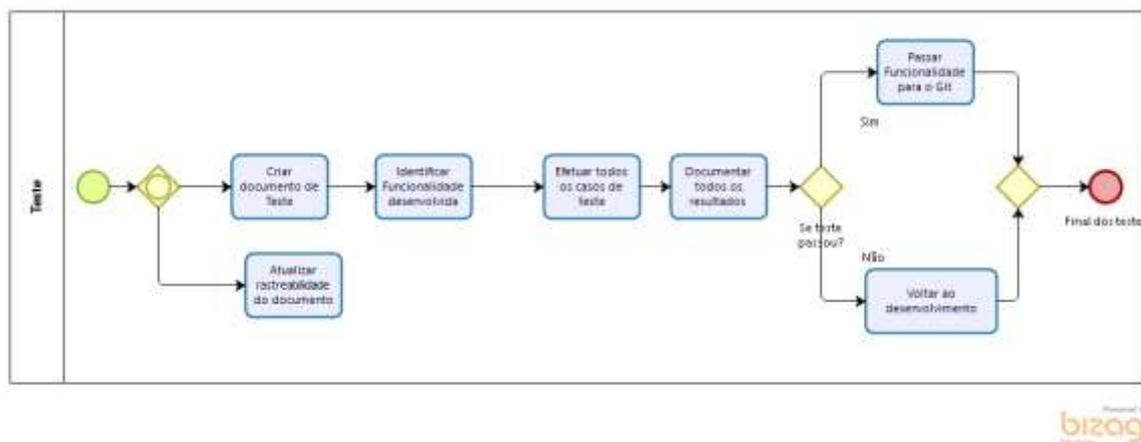
Fonte: Autor do trabalho.

Figura 3 - Modelo Ágil com cliente implícito Subprocesso Desenvolvimento.



Fonte: Autor do Trabalho.

Figura 4 - Modelo Ágil com cliente implícito Subprocesso Teste.



Fonte: Autor do Trabalho.

#### 4. Descrição das Atividades

Nesta sessão veremos a descrição das atividades, eventos e artefatos mostrados nos processos mostrados nas Figuras 1, 2, 3 e 4, será detalhado na mesma ordem de apresentação do modelo, mostrado em uma lista, além de incluir ferramentas que podem ser úteis no processo. Cada atividade, artefato ou evento mostrado anteriormente será listado e estarão destacados no texto em negrito.

- **Solicitação do cliente:** O evento de start do processo, ocorre quando o cliente envia uma solicitação para a equipe.
- **Análise de Solicitação e Viabilidade:** Os responsáveis fazem uma análise da solicitação e a viabilidade para saber se o sistema solicitado é possível de ser feito, caso seja então a solicitação passa para uma equipe que estará responsável por desenvolver o sistema.
- **Quando for necessário um novo time ou membro:** O evento pode ocorrer a cada início de ano, 6 meses, ou até mesmo com maior frequência, sempre que acontecer o evento começa e a parte do processo a seguir é executada.
- **Seleção de Times:** Quando for necessário que novos times ou membros sejam selecionados, então a empresa abre um processo seletivo da melhor forma possível para a empresa.
- **Treinamento dos novos times:** Os novos membros devem passar por um treinamento, incluindo as tecnologias que a empresa usa, como o trello, a tecnologias de desenvolvimento, Git, sobre o processo, os documentos, e uma um tempo para conhecer o sistema.
- **Sprint:** Evento que ocorre logo após os treinamentos, os membros começaram a participar efetivamente das sprints.
- **Criação do Product Backlog:** Os membros do time irão se reunir para decidir quais são as principais atividades que serão feitas nas Sprints e criar um documento, ou usar um quadro do trello para comportar todas as atividades que serão realizadas.
- **Início da Sprint:** Evento que marca o início das Sprints enquanto os times estão alocados aos projetos, as atividades após o evento podem ser refeitas em qualquer sprint e podem ser feitas em paralelas com outras atividades.

- **Reunião do Início da Sprint:** Em todo início da Sprint deve haver uma reunião entre os membros da equipe com os responsáveis, para decidir quais serão as atividades que serão feitas durante a Sprint, incluindo fazer a estimativa de tempo que cada atividade levará para ser concluída durante a Sprint, essa estimativa será feita através do Planning Poker, além se que é recomendado fazer as atividades com maior prioridade primeiro.
- **Fazer Estudo do público alvo:** Fazer um levantamento de quem será o público alvo da aplicação, através de pesquisas de mercado, ou até mesmo com questionário para o público, ou apenas observando o comportamento dos possíveis usuários.
- **Análise do processo existente:** Estudo de como é feito o processo do sistema que quer desenvolver, fazer um acompanhamento por observação, questionário ou entrevista com os possíveis usuários.
- **Realizar pesquisa de sistema existentes:** Fazer uma pesquisa de sistemas que já se propõe a fazer algo similar, encontrar esses sistemas pela internet, ou dependendo do tipo da plataforma que o sistema será desenvolvido, deve ser feita uma pesquisa sobre os concorrentes, encontrar o que os usuários sentem falta e o que precisa melhorar.
- **Sintetizar Informações:** Os dados obtidos das entrevistas, questionários, pesquisas e até mesmo observações, devem ser colocados em relatórios e planilhas, para organizar e deixar as informações mais claras e de compreensão.
- **Documentos Preliminares:** Um documento com as informações das entrevistas e as respostas dos questionários, essas informações vão conter, os requisitos funcionais, requisitos não funcionais, regras de negócio e até as histórias de usuário.
- **Elaboração do documento de requisitos:** Todas as informações do sistema estão neste documento, os requisitos, regras de negócio, diagramas, descrições e se a equipe achar necessário até mesmo histórias de usuário.
  - **Transformar requisitos de usuário, em requisitos de sistema:** Os dados colhidos nas entrevistas e questionários agora passaram a ser os requisitos do sistema, os membros da equipe responsáveis por isso devem tomar bastante cuidado em diferenciar os requisitos funcionais de requisitos não funcionais e regras de negócios, se caso algumas delas se mostrar de difícil entendimento, deve contatar o cliente para tirar as dúvidas, ou se caso for um requisito feito por um membro anterior, caso ele esteja disponível deve entrar em contato com ele, se caso for necessário pode se alterar o requisito.
  - **Criar as regras de negócios:** Com base nos documentos preliminares os responsáveis devem selecionar e identificar e separar as regras de negócio em uma sessão, caso ocorra de ser uma regra de negócio que foi escrita por outra pessoa, tente entrar em contato com o cliente ou com a pessoa que a escreveu.
  - **Criar requisitos não funcionais:** Com os requisitos de sistema, separe o que é requisito funcional e não funcional em sessões diferentes e todos devem conter os identificadores, caso precisem ser mudados, sempre lembrar de atualizar os identificadores, caso tenha dúvidas sobre requisitos existentes entrar em contato com o cliente ou com quem escreveu os requisitos.
  - **Criar Glossário do documento e a introdução:** O documento deve ter uma introdução do que se trata o sistema e como o documento é dividido, com todas as informações disponíveis já é possível criar, além de criar o glossário, uma sessão reservada a explicar os identificadores, abreviações e palavras pouco conhecidas.

- **Criar diagramas:** Nestas atividades todos os diagramas necessários ao sistema são criados, comumente serão os diagramas de caso de uso, de classe e de estado, mas se a empresa optar por mais pode utilizar quais quiser, todos serão colocados em uma sessão separados por subseções.
- **Criar legendas:** Já na sessão de descrição de caso de uso, uma legenda será criada para que seja mais facilmente visível as funcionalidades que já foram desenvolvidas ou que não desenvolveu, ou que foram parcialmente desenvolvidas, pode se utilizar cores para diferenciar.
- **Criar legendas:** Já na sessão de descrição de caso de uso, uma legenda será criada para que seja mais facilmente visível as funcionalidades que já foram desenvolvidas ou que não desenvolveu, ou que foram parcialmente desenvolvidas, pode se utilizar cores para diferenciar.
- **Criar descrições de caso de uso:** Cada caso de uso deve ter uma descrição correspondente, com todos os detalhes, como, nome, ator, descrição, requisitos e regras de negócio envolvidas, pré e pós condições e os fluxos principais, alternativos e de exceção, além de conter a legenda antes criada.
- **Atualizar o histórico de versionamento:** Sempre que o documento for alterado o histórico deve ser atualizado, com quem atualizou, quando, e quais foram as alterações.
- **Atualizar a Sessão de rastreamento:** O documento deve mostrar todos os documentos que possuem relação com ele, caso tenha, deve ser colocado em uma tabela o identificador do documento e fazer um link externo, o documento de requisitos por exemplo deve ter o identificador DRE junto a numeração corresponde ao documento como por exemplo 001 e assim por diante. Cada documento vai ter um identificador próprio e a numeração não deve ter relação com nenhum outro identificador.
- **Documento de requisitos:** Documento que contém os requisitos funcionais, não funcionais, regras de negócios, histórias de usuário, diagramas e descrições de casos de uso, será utilizado em quase todo o processo.
- **Validação do documento:** Uma reunião entre os membros da equipe é feita para validar o documento, após aos supervisores e ao cliente, caso precise de mudanças, essas devem ser feitas o quanto antes.
- **Desenvolvimento:** Com o documento de requisitos o mais completo possível, devem começar com as atividades de desenvolvimento nas sprints, mas sempre que encontrar algo que precise ser acrescentado ou mudado deve se atualizar o documento nas atividades anteriores.
  - **Identificar Funcionalidades:** Os desenvolvedores devem identificar todas as funcionalidades que precisam ser desenvolvidas, para então decidir entre os desenvolvedores qual ficará com quais funcionalidades durante a sprint.
  - **Pesquisar por soluções mais otimizadas que possam contribuir para o desenvolvimento:** Antes de começar a desenvolver, seria bem recomendado fazer pesquisas, para ver o que já existe, talvez algoritmos altamente usados para resolver os problemas, ou que sejam problemas parecidos que possam ser aproveitados na aplicação para agilizar o desenvolvimento.

- **Desenvolver Funcionalidade:** Com a pesquisa feita, então o desenvolvedor responsável deve começar a desenvolver a funcionalidade com base em suas descrições, após a funcionalidade está completa então deve passar para os testes.
- **Nova Funcionalidade:** É o arquivo ou classe com os códigos das novas funcionalidades.
- **Testes:** Os testes são uma parte importante, aqui algum membro da equipe que não necessariamente seja o desenvolvedor, deve realizar uma sequência de testes documentados para determinar se as funcionalidades estão fazendo o que foi especificado na documentação.
  - **Atualizar rastreabilidade do documento:** A seção do documento que faz referência aos documentos que o documento de teste faz referência, lembrando de diferenciar os identificadores por os respectivos documentos, como por exemplo documento de requisitos com o identificador DRE ou documento de teste DTE, e colocando a numeração correspondente ao documento e após deve linkar os documentos através de um link externo.
  - **Criar documentos de testes:** São os documentos mostra o passo a passo que o testador deve fazer para identificar se o sistema está fazendo o que deve fazer em todos os casos possíveis, além de documentar quem fez o teste, o resultado e caso tenha, uma solução ou sugestão, ou se a funcionalidade realmente faz o que deveria de acordo com as especificações.
  - **Identificar as funcionalidades desenvolvidas:** O testador deve verificar quais foram as funcionalidades desenvolvidas o quanto antes após seu desenvolvimento e fazer os testes.
  - **Efetuar todos os casos de teste:** Os casos de testes foram desenvolvidos com base no documento de requisitos, o testador deve realizar todos os casos para determinar se a funcionalidade faz ou não o que deve.
  - **Documentar todos os resultados:** Com os casos de teste feitos, o testador deve documentar todos os resultados encontrados no documento de teste.
  - **Passar funcionalidade para o Git:** Se caso a funcionalidade estiver funcionando corretamente ela deve ser colocada no repositório da empresa, para que todos os membros da equipe possam ter acesso e ter o controle sobre o desenvolvimento e marcar em todas as legendas das documentações, como completa, ou parcialmente desenvolvida, caso só uma parte dela seja desenvolvida.
  - **Voltar ao desenvolvimento:** O testador deve retornar a funcionalidade para lista de desenvolvimento, e marcar em todas as legendas nas documentações, que aquela funcionalidade ou falhou ou está incompleta.
- **Documento de teste:** Documento onde terá todos os casos de teste e os resultados dos testes realizados em cada funcionalidade, além de conter os nomes de quem realizou os testes e os resultados.
- **Reunião para controle de mudanças:** Sempre que alguma mudança deve ser feita uma reunião com a equipe e se possível com alguém que seja responsável pelo sistema, para discutir sobre as mudanças necessárias e saber os impactos que causará no sistema e se é viável ou não.
- **Criar relatório de mudanças:** Um membro da equipe deve criar um documento para identificar quem fez a mudança, por que fez a mudança e quando, sempre que algo for mudado deve ser adicionado a um relatório.

- **Relatório de mudanças:** Documento com todas as mudanças detalhadas, mostrando o que foi mudado, por quem, justificativa e quando a mudança foi feita, o documento sempre deve se manter atualizado, pois ele deve ser estudado por novos membros das equipes quando entrarem, para entender o motivo de determinadas escolhas de membros anteriores.
- **Reunião com times passados:** Quando os dados que estão disponíveis não forem o suficiente e membros anteriores estiverem disponíveis para se reunir, sempre é importante ir até eles para tirar dúvidas que persistem na documentação ou no código.
- **Relatório de reunião:** Para que as informações não se percam, um relatório deve ser escrito, contendo as dúvidas e as respostas obtidas.
- **Reunião de encerramento:** Uma reunião geral, para mostrar o que foi feito durante a sprint, o que foi aprendido e o que pode ser melhorado para a próxima sprint.
- **Estruturar documento de encerramento:** Elaborar documento que contenha as principais informações de como o sistema está ao final do estágio, e o que precisa ser feito, detalhadamente para que os próximos times agilizem o processo de desenvolvimento.
- **Documento de Encerramento:** Documento com informações sobre como foi que o sistema ficou ao término das Sprints, o que falta fazer, o que ficou incompleto, o que é mais importante, onde podem achar as informações necessárias.
- **Reunião Final:** Reunião de encerramento geral, onde serão apresentados os resultados e o que ficou por fazer.

Para esse processo é recomendado usar o trello, uma ferramenta de gestão de tarefas, assim a equipe saberá o que tem que fazer e o que cada um já faz, e o que estão fazendo, é bem simples de se usar e basta ter uma conta no google para usar.

Outra ferramenta são as disponibilizadas pelo google drive, como o docs, assim todos terão acesso aos documentos mais rapidamente, além de poder fazer apresentações e ter planilhas para o uso da equipe quando necessário.

Por fim o uso do Git para fazer a gestão do sistema, assim mantendo o controle da versão e que todos os membros da equipe tenham acesso aos mesmos códigos.

As entrevistas, questionários e observações ficam a critério dos desenvolvedores, pois eles variam facilmente conforme o sistema solicitado, porém ao final deste documento terá em Apêndice um exemplo de um documento de requisito completo usado por este processo, além do documento de teste e um modelo de relatório de mudanças.

## 5. Referências

CAVALCANTE, Mirlane Beserra. Proposta de um modelo baseado na combinação de metodologias ágeis de desenvolvimento de software. 2018. 115 f. Monografia (Graduação em Engenharia de Software) - Campus de Russas, Universidade Federal do Ceará, Russas, 2018.

MARTINS, Luiz Eduardo Galvão. Uma metodologia de elicitação de requisitos de software baseada na teoria da atividade. 2001. 170p. Tese (doutorado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação, Campinas, SP. Disponível em: <<http://www.repositorio.unicamp.br/handle/REPOSIP/260233>>. Acesso em: 17 ago. 2020.

MPS.BR - Melhoria de Processo do Software Brasileiro: Guia Geral MPS de Software, SofTex, 2020. Disponível em: <<https://softex.br/mpsbr/guias/#guia-sw>>. Acesso em: 17 ago. 2020.

O QUE É CMMI?. ISDBRASIL. Disponível em: <<http://www.isdbrasil.com.br/o-que-e-cmmi.php>>. Acesso em: 17 ago. 2020.

6. Apêndices

**APÊNDICE 1 - Exemplo de documento de requisitos**

(Imagem representando a empresa)

## Documento de Requisitos

Projeto: [*Nome do Projeto*]

Russas - Ceará

27/03/2021

## Sumário

<b><u>1. INTRODUÇÃO</u></b>	<b><u>5</u></b>
<u>1.1. Visão geral do documento</u>	<u>5</u>
<u>1.2. Identificação dos requisitos</u>	<u>5</u>
<u>1.3. Prioridades dos requisitos</u>	<u>5</u>
<b><u>2. RASTREAMENTO DE DOCUMENTAÇÃO</u></b>	<b><u>5</u></b>
<b><u>3. DESCRIÇÃO GERAL DO SISTEMA</u></b>	<b><u>5</u></b>
<u>3.1. Sistema</u>	<u>5</u>
<u>3.2. Stakeholders</u>	<u>6</u>
<b><u>4. LISTAGEM DOS REQUISITOS</u></b>	<b><u>6</u></b>
<u>4.1 Regras de Negócio</u>	<u>6</u>
<u>4.2. Estórias de Usuários</u>	<u>6</u>
<u>Requisitos Funcionais</u>	<u>7</u>
<u>4.3. Requisitos Não-Funcionais</u>	<u>7</u>
<b><u>5. MODELAGEM DO SISTEMA</u></b>	<b><u>7</u></b>
<b><u>6. DESCRIÇÃO DE CASO DE USO</u></b>	<b><u>7</u></b>

## HISTÓRICO DE ALTERAÇÕES

*[Ao final de cada modificação no documento o histórico de versão deve ser atualizado, a mudança de versão nem sempre é necessária, apenas quando a alteração for grande o suficiente para isso, além disso manter a prática de manter as linhas alternadas]*

Data	Versão	Descrição	Autor	Alterações
------	--------	-----------	-------	------------

--	--	--	--	--

## GLOSSÁRIO

*[Todos os termos pouco conhecidos e que podem ser estranhos, abreviações e nomenclaturas devem ser colocadas na tabela juntamente com sua descrição como mostrado no exemplo a seguir]*

Segue abaixo a tabela de termos e suas respectivas descrições usadas no sistema:

Termo	Descrição
Requisitos funcionais	Características e ações que o sistema deve realizar para atender a necessidade do cliente.
Requisitos funcionais não	Descrevem atributos e restrições que o sistema deve ter.
<i>Stakeholders</i>	Pessoas ou organizações que serão afetadas pelo sistema e que, direta ou indiretamente, têm influência sobre os requisitos.
<i>Templates</i>	Modelos de documentos que indicam o conteúdo necessário. A Engenharia de <i>Software</i> utiliza-os com muita frequência para sugerir padrões.

## ● 1. INTRODUÇÃO

*[Está seção do documento é destinada a introdução sobre o que se trata o documento de requisitos, essa sessão ainda será dividida em outras três subseções que serão mais específicas]*

### ○ 1.1. Visão geral do documento

*[Nesta subseção deve ser mostrado como o documento está dividido]*

### ○ 1.2. Identificação dos requisitos

*[Nesta subseção deve ser mostrado como será feito a identificação dos requisitos e explica cada uma delas]*

### ○ 1.3. Prioridades dos requisitos

*[Nesta subseção deve ser apresentado qual será a prioridade dos requisitos e qual a definição de cada prioridade]*

## ● 2. RASTREAMENTO DE DOCUMENTAÇÃO

*[Nesta seção deve ser colocado todos os documentos que serviram de referência direta para a criação deste documento, colocando o identificador do documento, junto com um link externo]*

## ● 3. DESCRIÇÃO GERAL DO SISTEMA

*[Está seção será dividida em duas subseções]*

### ○ 3.1. Sistema

*[Nesta subseção deve ser feita uma descrição completa sobre o sistema que está sendo desenvolvido, informações como para o que serve, funcionalidades de forma geral, entre outras informações]*

### ○ 3.2. Stakeholders

*[Nesta subseção deve ser descrito quem são todos os Stakeholders do sistema]*

## ● 4. LISTAGEM DOS REQUISITOS

*[Esta seção será dividida em três subseções]*

#### ○ 4.1 Regras de Negócio

*[Nesta subseção as regras de negócio devem ser identificadas e descritas de forma legível]*

ID Regra de Negócio	Título	Descrição

#### ○ 4.2. Estórias de Usuários

*[Nesta subseção as estórias de usuário devem ser colocadas em quadros, especificados, identificados, claros e coesos]*

<b>ID:</b>
<p style="text-align: center;"><b>Título: “”</b></p> <p><b>Descrição:</b></p> <p><b>Prioridade:</b></p>
<b>Critérios de Aceitação:</b>

<b>Testes a Realizar:</b>

**Requisitos Funcionais**

*[Nesta subseção os requisitos funcionais devem ser identificados e descritos de forma legível, coesa e coerente]*

Identificador	Título	Prioridade	Descrição
---------------	--------	------------	-----------

--	--	--	--

**4.3. Requisitos Não-Funcionais**

*[Nesta subseção os requisitos não-funcionais devem ser identificados e descritos conforme as necessidades, legíveis, entendíveis e coerentes]*

Identificador	Título	Descrição
---------------	--------	-----------

--	--	--

--	--	--

- **5. MODELAGEM DO SISTEMA**

*[Nesta seção estarão todos os diagramas feitos para modelar o sistema, cada diagrama será uma subseção desta]*

- **6. DESCRIÇÃO DE CASO DE USO**

*[Nesta seção cada caso de uso deve ser descrito exatamente da forma que ele deve se comportar em todos os cenários possíveis]*

O nível de desenvolvimento do componente é exibido na legenda abaixo:



**Componente desenvolvido**



**Componente em desenvolvimento**



**Componente desenvolvido, mas apresentou falhas**



**Componente ainda não desenvolvido**



**Situação não identificada**

Lista de casos de uso descritos:



Nome	
Ator Primário	
Sumário	
Pontos de Extensão	
Regras de Negócio	
Estórias de Usuário	
Requisitos Funcionais	
Pré-condições	
Pós-condições	
Fluxo Principal	
Fluxo Alternativo	
Fluxo de Exceção	

## APÊNDICE 1 - Exemplo de documento de requisitos

(imagem da empresa)

Relatório de Mudanças (*nome da empresa/sistema*)

Projeto: [*Nome do Projeto*]

Relatório de Mudanças

Nome	Data	Mudança	Motivo	Rastreamento de Documentação
Lucas Rafael	15/01/2021	A regra de negócio [RN013] foi removida.	A regra de negócio [RN008] já engloba a ocasião da regra de negócio [RN013].	[DR001]
Lucas Rafael	15/01/2021	O ator professor foi removido da documentação.	Pois foi se notado que a parte que condizia com o ator professor já não fazia mais parte do que o sistema deveria fazer.	[DR001], [DMU001], [DTE001]
Lucas Rafael	15/01/2021	O requisito funcional [RF007] foi alterado.	O requisito se mostrava ambíguo e repetia o que o requisito funcional [RF003] fazia, porém de forma confusa.	[DR001]

**Apêndice 5 – Modelo de Processo Ágil Simplificado com Cliente.**

# Modelo de Processo Ágil Simplificado com Clientes

Modelo proposto por: Lucas Rafael Alves Oliveira

Março de 2021

**SUMÁRIO**

**Introdução**    **3**

**Objetivo**    **3**

**Modelo Gráfico**    **3**

**Descrição das Atividades**    **4**

**Referências**    **6**

**Apêndices**    **7**

## 1. Introdução

Pensando em empresas que precisam de velocidade em seus processos, o modelo de processo ágil aqui proposto, foi pensado para equipes com experiência intermediária e com estabilidade em suas equipes, o processo leva em consideração os níveis de qualidade do CMMI e MPS Br, porém o MPS Br será mais amplamente abordado durante o processo.

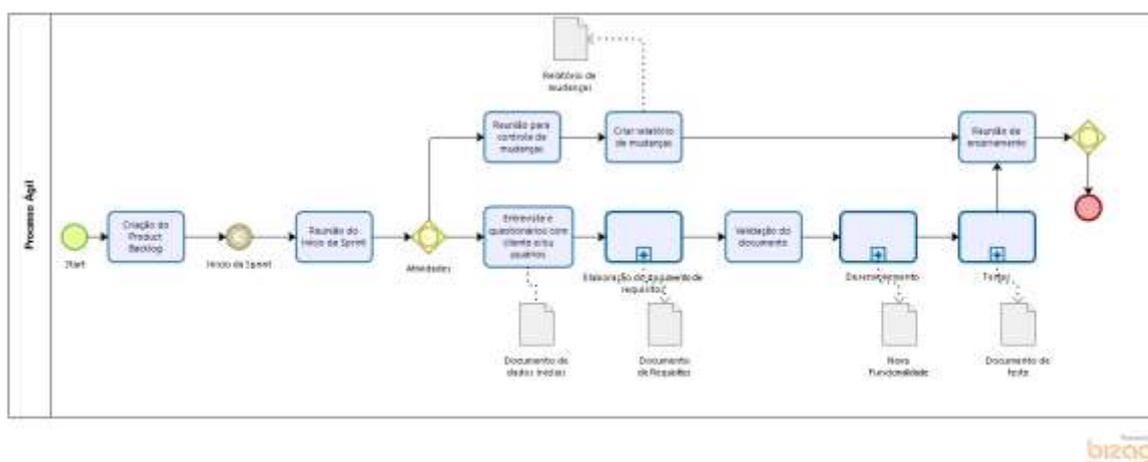
## 2. Objetivo

O processo tem como objetivo dar aos usuários do guia um processo ágil e simplificado, mas que ainda assim possua os quesitos de qualidade proposto pelo CMMI e MPS Br. Assim fornecendo agilidade, facilidade e qualidade.

## 3. Modelo Gráfico

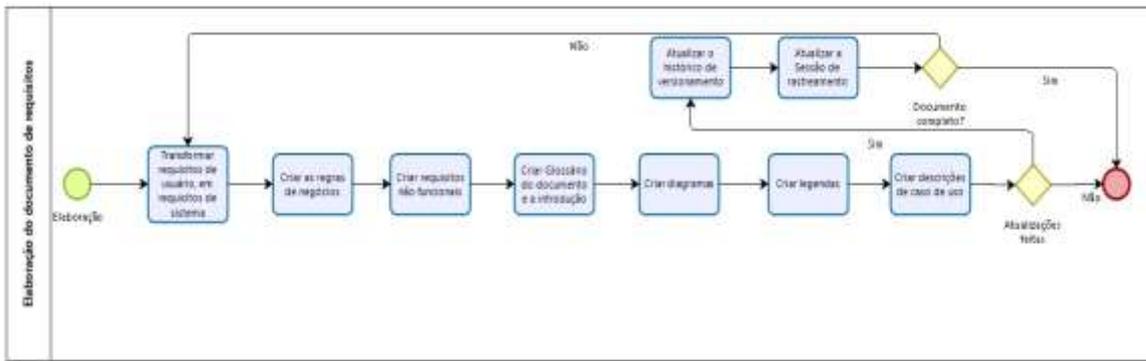
Nesta sessão iremos ver o modelo gráfico do processo ágil com cliente implícito, focando na frequência de novos membros na equipe, como é mostrado nas Figuras 1, 2, 3 e 4.

Figura 1 - Processo de Ágil Simplificado com Cliente.



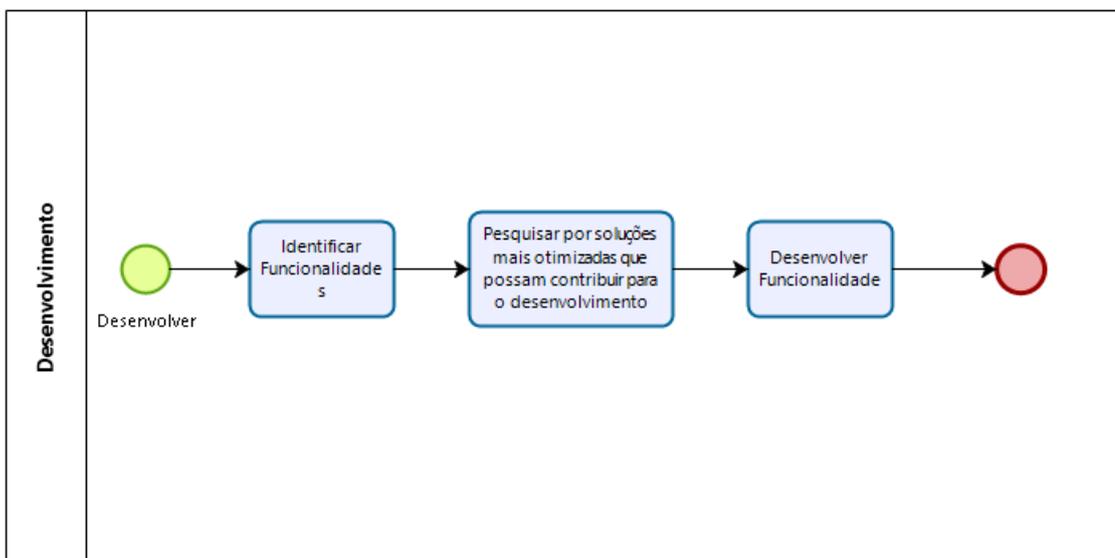
Fonte: Autor do trabalho.

Figura 2 - Modelo Ágil Simplificado com Cliente Subprocesso Elaboração do Documento de Requisitos.



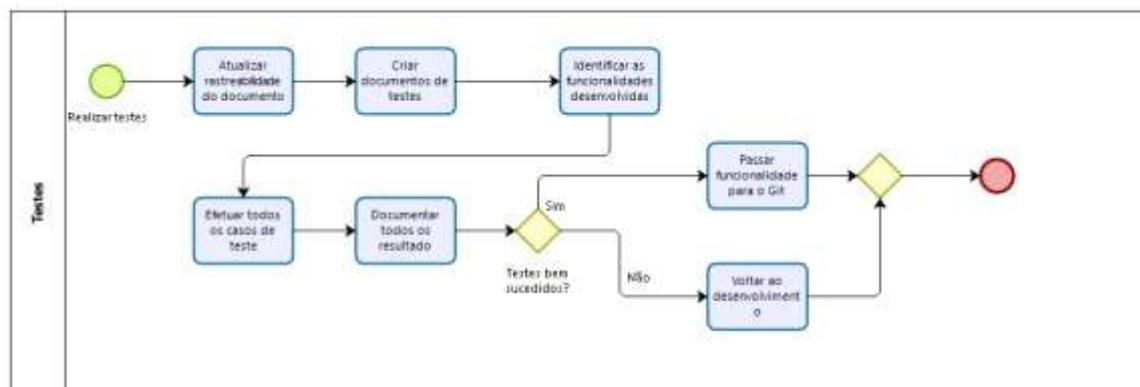
Fonte: Autor do Trabalho.

Figura 3 - Modelo Ágil Simplificado com Cliente Subprocesso Desenvolvimento.



Fonte: Autor do Trabalho.

Figura 4 - Modelo Ágil Simplificado com Cliente Subprocesso Testes



Powered by  
bizagi

Fonte: Autor do Trabalho.

#### 4. Descrição das Atividades

Nesta sessão veremos a descrição das atividades, eventos e artefatos mostrados nos processos mostrados nas Figuras 1, 2, 3 e 4, será detalhado na mesma ordem de apresentação do modelo, mostrado em uma lista, além de incluir ferramentas que podem ser úteis no processo. Cada atividade, artefato ou evento mostrado anteriormente será listado e estarão destacados no texto em negrito.

- **Start:** Evento que dá início ao processo, quando a equipe tem uma ideia de sistema para desenvolver
- **Criação do Product Backlog:** Os membros do time irão se reunir para decidir quais são as principais atividades que serão feitas nas Sprints e criar um documento, ou usar um quadro do trello para comportar todas as atividades que serão realizadas.
- **Início da Sprint:** Evento que marca o início das Sprints enquanto os times estão alocados aos projetos, as atividades após o evento podem ser refeitas em qualquer sprint e podem ser feitas em paralelas com outras atividades.
- **Reunião do Início da Sprint:** Em todo início da Sprint deve haver uma reunião entre os membros da equipe com os responsáveis, para decidir quais serão as atividades que serão feitas durante a Sprint, incluindo fazer a estimativa de tempo que cada atividade levará para ser concluída durante a Sprint, essa estimativa será feita através do Planning Poker, além se que é recomendado fazer as atividades com maior prioridade primeiro.
- **Entrevista e questionários com cliente e/ou usuários:** Nesta atividade membros das equipes devem ver como é o processo que os clientes ou usuários fazem sem o sistema e também fazer entrevistas e questionários sobre as funcionalidades que os clientes e usuários desejam que seja implementada.
- **Documento de dados iniciais:** Um documento com as informações das entrevistas e as respostas dos questionários, essas informações vão conter, os requisitos funcionais, requisitos não funcionais, regras de negócio e até as estórias de usuário.
- **Elaboração do documento de requisitos:** Todas as informações do sistema estão neste documento, os requisitos, regras de negócio, diagramas, descrições e se a equipe achar necessário até mesmo estórias de usuário.

- **Transformar requisitos de usuário, em requisitos de sistema:** Os dados colhidos nas entrevistas e questionários agora passaram a ser os requisitos do sistema, os membros da equipe responsáveis por isso devem tomar bastante cuidado em diferenciar os requisitos funcionais de requisitos não funcionais e regras de negócios, se caso algumas delas se mostrar de difícil entendimento, deve contatar o cliente para tirar as dúvidas, ou se caso for um requisito feito por um membro anterior, caso ele esteja disponível deve entrar em contato com ele, se caso for necessário pode se alterar o requisito.
- **Criar as regras de negócios:** Com base nos documentos preliminares os responsáveis devem selecionar e identificar e separar as regras de negócio em uma sessão, caso ocorra de ser uma regra de negócio que foi escrita por outra pessoa, tente entrar em contato com o cliente ou com a pessoa que a escreveu.
- **Criar requisitos não funcionais:** Com os requisitos de sistema, separe o que é requisito funcional e não funcional em sessões diferentes e todos devem conter os identificadores, caso precisem ser mudados, sempre lembrar de atualizar os identificadores, caso tenha dúvidas sobre requisitos existentes entrar em contato com o cliente ou com quem escreveu os requisitos.
- **Criar Glossário do documento e a introdução:** O documento deve ter uma introdução do que se trata o sistema e como o documento é dividido, com todas as informações disponíveis já é possível criar, além de criar o glossário, uma sessão reservada a explicar os identificadores, abreviações e palavras pouco conhecidas.
- **Criar diagramas:** Nestas atividades todos os diagramas necessários ao sistema são criados, comumente serão os diagramas de caso de uso, de classe e de estado, mas se a empresa optar por mais pode utilizar quais quiser, todos serão colocados em uma sessão separados por subseções.
- **Criar legendas:** Já na sessão de descrição de caso de uso, uma legenda será criada para que seja mais facilmente visível as funcionalidades que já foram desenvolvidas ou que não desenvolveu, ou que foram parcialmente desenvolvidas, pode se utilizar cores para diferenciar.
- **Criar descrições de caso de uso:** Cada caso de uso deve ter uma descrição correspondente, com todos os detalhes, como, nome, ator, descrição, requisitos e regras de negócio envolvidas, pré e pós condições e os fluxos principais, alternativos e de exceção, além de conter a legenda antes criada.
- **Atualizar o histórico de versionamento:** Sempre que o documento for alterado o histórico deve ser atualizado, com quem atualizou, quando, e quais foram as alterações.
- **Atualizar a Sessão de rastreamento:** O documento deve mostrar todos os documentos que possuem relação com ele, caso tenha, deve ser colocado em uma tabela o identificador do documento e fazer um link externo, o documento de requisitos por exemplo deve ter o identificador DRE junto a numeração corresponde ao documento como por exemplo 001 e assim por diante. Cada documento vai ter um identificador próprio e a numeração não deve ter relação com nenhum outro identificador.
- **Documento de requisitos:** Documento que contém os requisitos funcionais, não funcionais, regras de negócios, histórias de usuário, diagramas e descrições de casos de uso, será utilizado em quase todo o processo.

- **Validação do documento:** Uma reunião entre os membros da equipe é feita para validar o documento, caso precise de mudanças, essas devem ser feitas o quanto antes.
- **Desenvolvimento:** Com o documento de requisitos o mais completo possível, devem começar com as atividades de desenvolvimento nas sprints, mas sempre que encontrar algo que precise ser acrescentado ou mudado deve se atualizar o documento nas atividades anteriores.
  - **Identificar Funcionalidades:** Os desenvolvedores devem identificar todas as funcionalidades que precisam ser desenvolvidas, para então decidir entre os desenvolvedores qual ficará com quais funcionalidades durante a sprint.
  - **Pesquisar por soluções mais otimizadas que possam contribuir para o desenvolvimento:** Antes de começar a desenvolver, seria bem recomendado fazer pesquisas para ver o que já existe, talvez algoritmos altamente usados para resolver os problemas, ou que sejam problemas parecidos que possam ser aproveitados na aplicação para agilizar o desenvolvimento.
  - **Desenvolver Funcionalidade:** Com a pesquisa feita, então o desenvolvedor responsável deve começar a desenvolver a funcionalidade com base em suas descrições, após a funcionalidade está completa então deve passar para os testes, durante o desenvolvimento é importante que as linhas de código sejam escritas de forma clara e intuitiva, além de deixar algumas partes do código bem comentadas.
- **Nova Funcionalidade:** É o arquivo ou classe com os códigos das novas funcionalidades.
- **Testes:** Os testes são uma parte importante, aqui algum membro da equipe que não necessariamente seja o desenvolvedor, deve realizar uma sequência de testes documentados para determinar se as funcionalidades estão fazendo o que foi especificado na documentação.
  - **Atualizar rastreabilidade do documento:** A seção do documento que faz referência aos documentos que o documento de teste faz referência, lembrando de diferenciar os identificadores por os respectivos documentos, como por exemplo documento de requisitos com o identificador DRE ou documento de teste DTE, e colocando a numeração correspondente ao documento e após deve linkar os documentos através de um link externo.
  - **Criar documentos de testes:** São os documentos mostra o passo a passo que o testador deve fazer para identificar se o sistema está fazendo o que deve fazer em todos os casos possíveis, além de documentar quem fez o teste, o resultado e caso tenha, uma solução ou sugestão, ou se a funcionalidade realmente faz o que deveria de acordo com as especificações.
  - **Identificar as funcionalidades desenvolvidas:** O testador deve verificar quais foram as funcionalidades desenvolvidas o quanto antes após seu desenvolvimento e fazer os testes.
  - **Efetuar todos os casos de teste:** Os casos de testes foram desenvolvidos com base no documento de requisitos, o testador deve realizar todos os casos para determinar se a funcionalidade faz ou não o que deve.
  - **Documentar todos os resultados:** Com os casos de teste feitos, o testador deve documentar todos os resultados encontrados no documento de teste.
  - **Passar funcionalidade para o Git:** Se caso a funcionalidade estiver funcionando corretamente ela deve ser colocada no repositório da empresa, para que todos os membros da equipe possam ter acesso e ter o controle sobre o desenvolvimento e

marcar em todas as legendas das documentações, como completa, ou parcialmente desenvolvida, caso só uma parte dela seja desenvolvida.

- **Voltar ao desenvolvimento:** O testador deve retornar a funcionalidade para lista de desenvolvimento, e marcar em todas as legendas nas documentações, que aquela funcionalidade ou falhou ou está incompleta.
- **Documento de teste:** Documento onde terá todos os casos de teste e os resultados dos testes realizados em cada funcionalidade, além de conter os nomes de quem realizou os testes e os resultados.
- **Reunião para controle de mudanças:** Sempre que alguma mudança deve ser feita uma reunião com a equipe e se possível com alguém que seja responsável pelo sistema, para discutir sobre as mudanças necessárias e saber os impactos que causará no sistema e se é viável ou não.
- **Criar relatório de mudanças:** Um membro da equipe deve criar um documento para identificar quem fez a mudança, por que fez a mudança e quando, sempre que algo for mudado deve ser adicionado a um relatório.
- **Relatório de mudanças:** Documento com todas as mudanças detalhadas, mostrando o que foi mudado, por quem, justificativa e quando a mudança foi feita, o documento sempre deve se manter atualizado, pois ele deve ser estudado por novos membros das equipes quando entrarem, para entender o motivo de determinadas escolhas de membros anteriores.
- **Reunião de encerramento:** Uma reunião geral, para mostrar o que foi feito durante a sprint, o que foi aprendido e o que pode ser melhorado para a próxima sprint.

Para esse processo é recomendado usar o trello, uma ferramenta de gestão de tarefas, assim a equipe saberá o que tem que fazer e o que cada um já faz, e o que estão fazendo, é bem simples de se usar e basta ter uma conta no google para usar.

Outra ferramenta são as disponibilizadas pelo google drive, como o docs, assim todos terão acesso aos documentos mais rapidamente, além de poder fazer apresentações e ter planilhas para o uso da equipe quando necessário.

Por fim o uso do Git para fazer a gestão do sistema, assim mantendo o controle da versão e que todos os membros da equipe tenham acesso aos mesmos códigos.

As entrevistas, questionários e observações ficam a critério dos desenvolvedores, pois eles variam facilmente conforme o sistema solicitado, porém ao final deste documento terá em Apêndice um exemplo de um documento de requisito completo usado por este processo, além do documento de teste e um modelo de relatório de mudanças.

## 5. Referências

CAVALCANTE, Mirlane Beserra. Proposta de um modelo baseado na combinação de metodologias ágeis de desenvolvimento de software. 2018.115 f. Monografia (Graduação em Engenharia de Software) - Campus de Russas, Universidade Federal do Ceará, Russas, 2018.

MARTINS, Luiz Eduardo Galvão. Uma metodologia de elicitação de requisitos de software baseada na teoria da atividade. 2001. 170p. Tese (doutorado) - Universidade Estadual de Campinas, Faculdade de Engenharia Eletrica e de Computação, Campinas, SP. Disponível em: <<http://www.repositorio.unicamp.br/handle/REPOSIP/260233>>. Acesso em: 17 ago. 2020.

MPS.BR - Melhoria de Processo do Software Brasileiro: Guia Geral MPS de Software, SofTex, 2020. Disponível em: <<https://softex.br/mpsbr/guias/#guia-sw>>. Acesso em: 17 ago. 2020.

O QUE É CMMI?. ISDBRASIL. Disponível em: <<http://www.isdbrazil.com.br/o-que-e-cmmi.php>>. Acesso em: 17 ago. 2020.

**6.** Apêndices**APÊNDICE 1 - Exemplo de documento de requisitos**

(Imagem representando a empresa)

## Documento de Requisitos

Projeto: [*Nome do Projeto*]

Russas - Ceará

26/03/2021

## Sumário

<b><u>1. INTRODUÇÃO</u></b>	<b><u>5</u></b>
<u>1.1. Visão geral do documento</u>	<u>5</u>
<u>1.2. Identificação dos requisitos</u>	<u>5</u>
<u>1.3. Prioridades dos requisitos</u>	<u>5</u>
<b><u>2. RASTREAMENTO DE DOCUMENTAÇÃO</u></b>	<b><u>5</u></b>
<b><u>3. DESCRIÇÃO GERAL DO SISTEMA</u></b>	<b><u>5</u></b>
<u>3.1. Sistema</u>	<u>5</u>
<u>3.2. Stakeholders</u>	<u>6</u>
<b><u>4. LISTAGEM DOS REQUISITOS</u></b>	<b><u>6</u></b>
<u>4.1 Regras de Negócio</u>	<u>6</u>
<u>4.2. Estórias de Usuários</u>	<u>6</u>
<u>Requisitos Funcionais</u>	<u>7</u>
<u>4.3. Requisitos Não-Funcionais</u>	<u>7</u>
<b><u>5. MODELAGEM DO SISTEMA</u></b>	<b><u>7</u></b>
<b><u>6. DESCRIÇÃO DE CASO DE USO</u></b>	<b><u>7</u></b>

## HISTÓRICO DE ALTERAÇÕES

*[Ao final de cada modificação no documento o histórico de versão deve ser atualizado, a mudança de versão nem sempre é necessária, apenas quando a alteração for grande o suficiente para isso, além disso manter a prática de manter as linhas alternadas]*

Data	Versão	Descrição	Autor	Alterações
------	--------	-----------	-------	------------

--	--	--	--	--

## GLOSSÁRIO

*[Todos os termos pouco conhecidos e que podem ser estranhos, abreviações e nomenclaturas devem ser colocadas na tabela juntamente com sua descrição como mostrado no exemplo a seguir]*

Segue abaixo a tabela de termos e suas respectivas descrições usadas no sistema:

Termo	Descrição
Requisitos funcionais	Características e ações que o sistema deve realizar para atender a necessidade do cliente.
Requisitos funcionais não	Descrevem atributos e restrições que o sistema deve ter.
<i>Stakeholders</i>	Pessoas ou organizações que serão afetadas pelo sistema e que, direta ou indiretamente, têm influência sobre os requisitos.
<i>Templates</i>	Modelos de documentos que indicam o conteúdo necessário. A Engenharia de <i>Software</i> utiliza-os com muita frequência para sugerir padrões.

## ● 1. INTRODUÇÃO

*[Esta seção do documento é destinada a introdução sobre o que se trata o documento de requisitos, essa sessão ainda será dividida em outras três subseções que serão mais específicas]*

### ○ 1.1. Visão geral do documento

*[Nesta subseção deve ser mostrado como o documento está dividido]*

### ○ 1.2. Identificação dos requisitos

*[Nesta subseção deve ser mostrado como será feito a identificação dos requisitos e explica cada uma delas]*

### ○ 1.3. Prioridades dos requisitos

*[Nesta subseção deve ser apresentado qual será a prioridade dos requisitos e qual a definição de cada prioridade]*

## ● 2. RASTREAMENTO DE DOCUMENTAÇÃO

*[Nesta seção deve ser colocado todos os documentos que serviram de referência direta para a criação deste documento, colocando o identificador do documento, junto com um link externo]*

## ● 3. DESCRIÇÃO GERAL DO SISTEMA

*[Esta seção será dividida em duas subseções]*

### ○ 3.1. Sistema

*[Nesta subseção deve ser feita uma descrição completa sobre o sistema que está sendo desenvolvido, informações como para o que serve, funcionalidades de forma geral, entre outras informações]*

### ○ 3.2. Stakeholders

*[Nesta subseção deve ser descrito quem são todos os Stakeholders do sistema]*

## ● 4. LISTAGEM DOS REQUISITOS

*[Esta seção será dividida em três subseções]*

#### ○ 4.1 Regras de Negócio

*[Nesta subseção as regras de negócio devem ser identificadas e descritas de forma legível]*

ID Regra de Negócio	Título	Descrição

#### ○ 4.2. Estórias de Usuários

*[Nesta subseção as estórias de usuário devem ser colocadas em quadros, especificados, identificados, claros e coesos]*

<b>ID:</b>
<p style="text-align: center;"><b>Título: “”</b></p> <p><b>Descrição:</b></p> <p><b>Prioridade:</b></p>
<b>Critérios de Aceitação:</b>

<b>Testes a Realizar:</b>

**Requisitos Funcionais**

*[Nesta subseção os requisitos funcionais devem ser identificados e descritos de forma legível, coesa e coerente]*

Identificador	Título	Prioridade	Descrição
---------------	--------	------------	-----------

--	--	--	--

**4.3. Requisitos Não-Funcionais**

*[Nesta subseção os requisitos não-funcionais devem ser identificados e descritos conforme as necessidades, legíveis, entendíveis e coerentes]*

Identificador	Título	Descrição
---------------	--------	-----------

--	--	--

--	--	--

- **5. MODELAGEM DO SISTEMA**

*[Nesta seção estarão todos os diagramas feitos para modelar o sistema, cada diagrama será uma subseção desta]*

- **6. DESCRIÇÃO DE CASO DE USO**

*[Nesta seção cada caso de uso deve ser descrito exatamente da forma que ele deve se comportar em todos os cenários possíveis]*

O nível de desenvolvimento do componente é exibido na legenda abaixo:



**Componente desenvolvido**



**Componente em desenvolvimento**



**Componente desenvolvido, mas apresentou falhas**



**Componente ainda não desenvolvido**



**Situação não identificada**

Lista de casos de uso descritos:



Nome	
Ator Primário	
Sumário	
Pontos de Extensão	
Regras de Negócio	
Estórias de Usuário	
Requisitos Funcionais	
Pré-condições	
Pós-condições	
Fluxo Principal	
Fluxo Alternativo	
Fluxo de Exceção	

## APÊNDICE 2 - Exemplo de documento de Relatório de Mudanças

(imagem da empresa)

Relatório de Mudanças (*nome da empresa/sistema*)

Projeto: [*Nome do Projeto*]

Relatório de Mudanças

Nome	Data	Mudança	Motivo	Rastreamento de Documentação
Lucas Rafael	15/01/2021	A regra de negócio [RN013] foi removida.	A regra de negócio [RN008] já engloba a ocasião da regra de negócio [RN013].	[DR001]
Lucas Rafael	15/01/2021	O ator professor foi removido da documentação.	Pois foi se notado que a parte que condizia com o ator professor já não fazia mais parte do que o sistema deveria fazer.	[DR001], [DMU001], [DTE001]
Lucas Rafael	15/01/2021	O requisito funcional [RF007] foi alterado.	O requisito se mostrava ambíguo e repetia o que o requisito funcional [RF003] fazia, porém de forma confusa.	[DR001]

**Apêndice 6 – Modelo de Processo Ágil Simplificado com cliente implícito.**

# Modelo de Processo Ágil Simplificado com Cliente Implícito

Modelo proposto por: Lucas Rafael Alves Oliveira

Março de 2021

**SUMÁRIO**

**Introdução**    **3**

**Objetivo**    **3**

**Modelo Gráfico**    **3**

**Descrição das Atividades**    **4**

**Referências**    **6**

**Apêndices**    **7**

## 1. Introdução

Pensando em empresas que precisam de velocidade em seus processos, o modelo de processo ágil aqui proposto, foi pensado para equipes com experiência intermediária e com estabilidade em suas equipes, o processo leva em consideração os níveis de qualidade do CMMI e MPS Br, porém o MPS Br será mais amplamente abordado durante o processo.

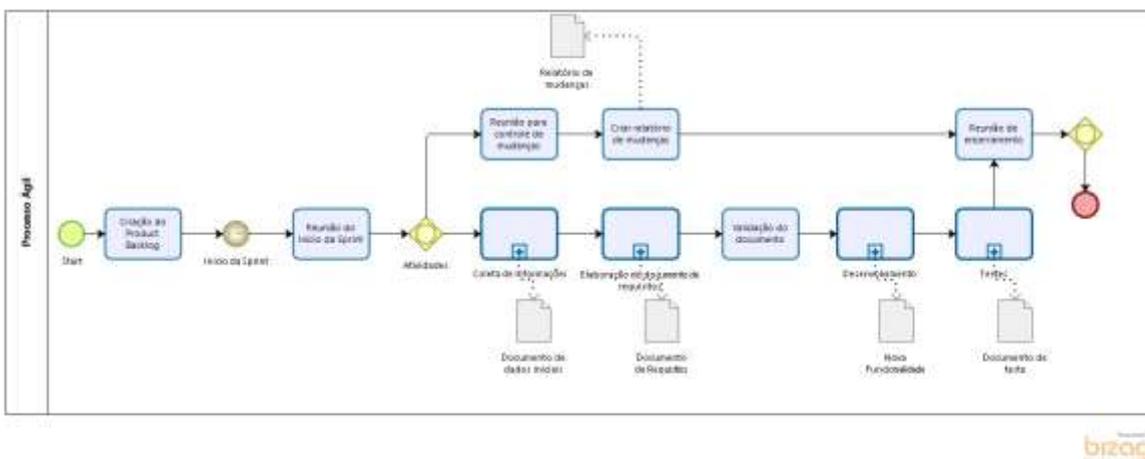
## 2. Objetivo

O processo tem como objetivo dar aos usuários do guia um processo ágil e simplificado, mas que ainda assim possua os quesitos de qualidade proposto pelo CMMI e MPS Br. Assim fornecendo agilidade, facilidade e qualidade.

## 3. Modelo Gráfico

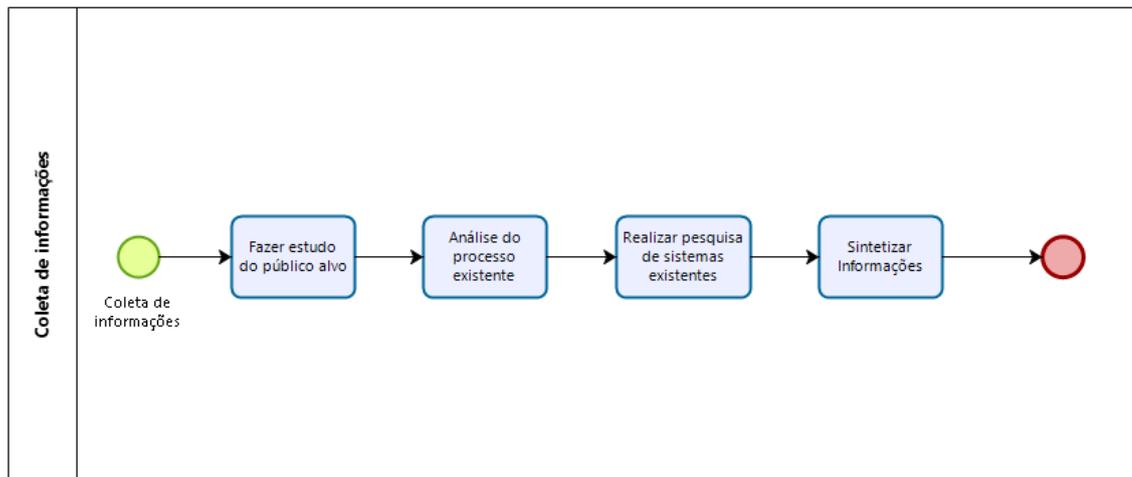
Nesta sessão iremos ver o modelo gráfico do processo ágil com cliente implícito, focando na frequência de novos membros na equipe, como é mostrado nas Figuras 1, 2, 3, 4 e 5.

Figura 1 - Processo de Ágil Simplificado com cliente implícito.



Fonte: Autor do trabalho.

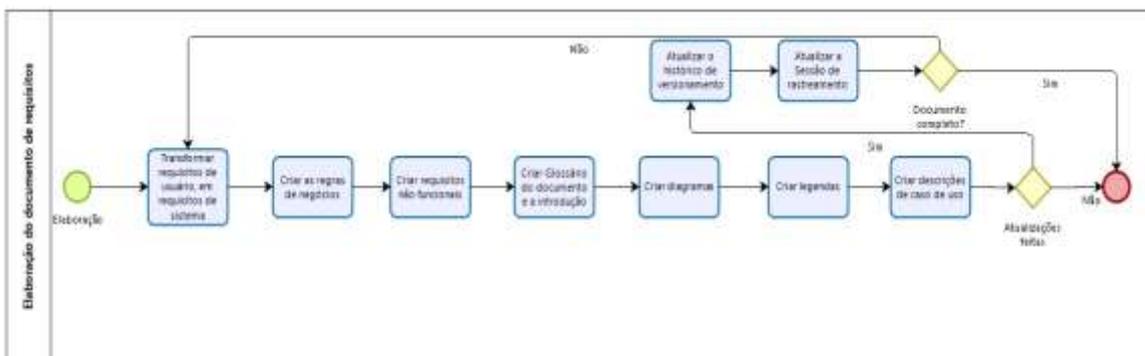
Figura 2 - Modelo Ágil Simplificado com cliente implícito Subprocesso Coletar Informações.



Powered by  
bizagi  
Modeler

Fonte: Autor do trabalho.

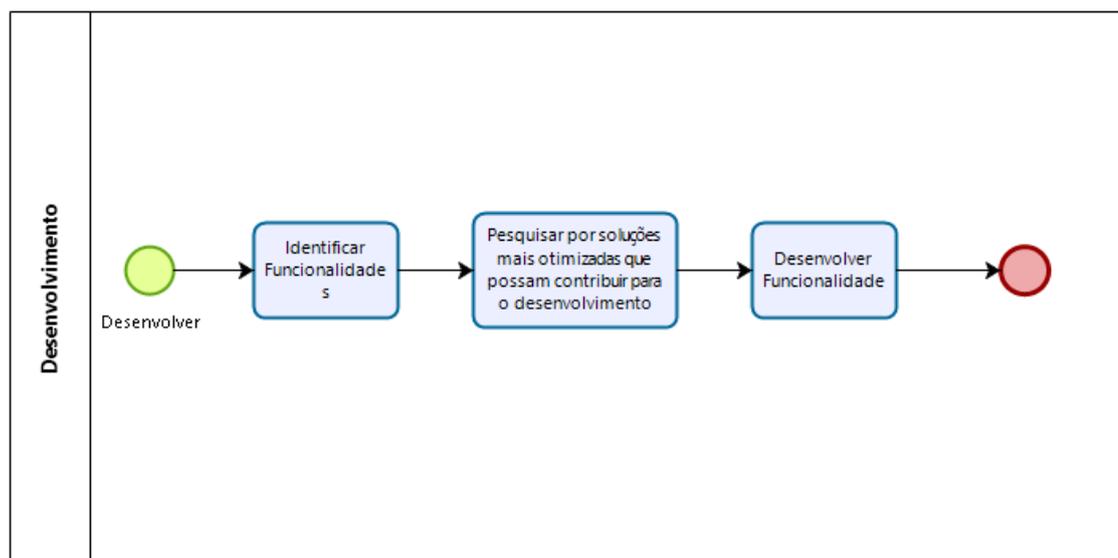
Figura 3 - Modelo Ágil Simplificado com cliente implícito Subprocesso Elaboração do Documento de Requisitos.



Powered by  
bizagi  
Modeler

Fonte: Autor do Trabalho.

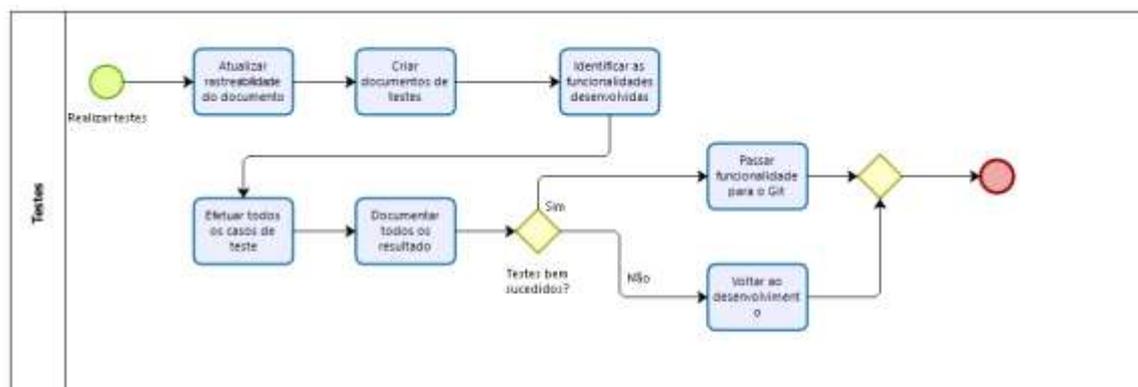
Figura 4 - Modelo Ágil Simplificado com cliente implícito Subprocesso Desenvolvimento.



Powered by  
bizagi  
Modeler

Fonte: Autor do Trabalho.

Figura 5 - Modelo Ágil Simplificado com cliente implícito Subprocesso Testes



Powered by  
bizagi  
Modeler

Fonte: Autor do Trabalho.

#### 4. Descrição das Atividades

Nesta sessão veremos a descrição das atividades, eventos e artefatos mostrados nos processos mostrados nas Figuras 1, 2, 3 e 4, será detalhado na mesma ordem de apresentação do modelo, mostrado em uma lista, além de incluir ferramentas que podem ser úteis no processo. Cada atividade, artefato ou evento mostrado anteriormente será listado e estarão destacados no texto em negrito.

- **Start:** Evento que dá início ao processo, quando a equipe tem uma ideia de sistema para desenvolver
- **Criação do Product Backlog:** Os membros do time irão se reunir para decidir quais são as principais atividades que serão feitas nas Sprints e criar um documento, ou usar um quadro do trello para comportar todas as atividades que serão realizadas.
- **Início da Sprint:** Evento que marca o início das Sprints enquanto os times estão alocados aos projetos, as atividades após o evento podem ser refeitas em qualquer sprint e podem ser feitas em paralelas com outras atividades.
- **Reunião do Início da Sprint:** Em todo início da Sprint deve haver uma reunião entre os membros da equipe com os responsáveis, para decidir quais serão as atividades que serão feitas durante a Sprint, incluindo fazer a estimativa de tempo que cada atividade levará para ser concluída durante a Sprint, essa estimativa será feita através do Planning Poker, além se que é recomendado fazer as atividades com maior prioridade primeiro.
- **Coleta de Informações:** Tarefa que engloba tudo que se refere às atividades para coleta de informações para iniciar a criação do documento de requisitos
  - **Fazer Estudo do público alvo:** Fazer um levantamento de quem será o público alvo da aplicação, através de pesquisas de mercado, ou até mesmo com questionário para o público, ou apenas observando o comportamento dos possíveis usuários.
  - **Análise do processo existente:** Estudo de como é feito o processo do sistema que quer desenvolver, fazer um acompanhamento por observação, questionário ou entrevista com os possíveis usuários.
  - **Realizar pesquisa de sistema existentes:** Fazer uma pesquisa de sistemas que já se propõe a fazer algo similar, encontrar esses sistemas pela internet, ou dependendo do tipo da plataforma que o sistema será desenvolvido, deve ser feita uma pesquisa sobre os concorrentes, encontrar o que os usuários sentem falta e o que precisa melhorar.
  - **Sintetizar Informações:** Os dados obtidos das entrevistas, questionários, pesquisas e até mesmo observações, devem ser colocados em relatórios e planilhas, para organizar e deixar as informações mais claras e de compreensão.
- **Documento de dados iniciais:** Um documento com as informações das entrevistas e as respostas dos questionários, essas informações vão conter, os requisitos funcionais, requisitos não funcionais, regras de negócio e até as histórias de usuário.
- **Elaboração do documento de requisitos:** Todas as informações do sistema estão neste documento, os requisitos, regras de negócio, diagramas, descrições e se a equipe achar necessário até mesmo histórias de usuário.
  - **Transformar requisitos de usuário, em requisitos de sistema:** Os dados colhidos nas entrevistas e questionários agora passaram a ser os requisitos do sistema, os membros da equipe responsáveis por isso devem tomar bastante cuidado em diferenciar os requisitos funcionais de requisitos não funcionais e regras de negócios, se caso algumas delas se mostrar de difícil entendimento, deve contatar o cliente para tirar as dúvidas, ou se caso for um requisito feito por um membro anterior, caso ele esteja disponível deve entrar em contato com ele, se caso for necessário pode se alterar o requisito.
  - **Criar as regras de negócios:** Com base nos documentos preliminares os responsáveis devem selecionar e identificar e separar as regras de negócio em uma sessão, caso

ocorra de ser uma regra de negócio que foi escrita por outra pessoa, tente entrar em contato com o cliente ou com a pessoa que a escreveu.

- **Criar requisitos não funcionais:** Com os requisitos de sistema, separe o que é requisito funcional e não funcional em sessões diferentes e todos devem conter os identificadores, caso precisem ser mudados, sempre lembrar de atualizar os identificadores, caso tenha dúvidas sobre requisitos existentes entrar em contato com o cliente ou com quem escreveu os requisitos.
  - **Criar Glossário do documento e a introdução:** O documento deve ter uma introdução do que se trata o sistema e como o documento é dividido, com todas as informações disponíveis já é possível criar, além de criar o glossário, uma sessão reservada a explicar os identificadores, abreviações e palavras pouco conhecidas.
  - **Criar diagramas:** Nestas atividades todos os diagramas necessários ao sistema são criados, comumente serão os diagramas de caso de uso, de classe e de estado, mas se a empresa optar por mais pode utilizar quais quiser, todos serão colocados em uma sessão separados por subseções.
  - **Criar legendas:** Já na sessão de descrição de caso de uso, uma legenda será criada para que seja mais facilmente visível as funcionalidades que já foram desenvolvidas ou que não desenvolveu, ou que foram parcialmente desenvolvidas, pode se utilizar cores para diferenciar.
  - **Criar descrições de caso de uso:** Cada caso de uso deve ter uma descrição correspondente, com todos os detalhes, como, nome, ator, descrição, requisitos e regras de negócio envolvidas, pré e pós condições e os fluxos principais, alternativos e de exceção, além de conter a legenda antes criada.
  - **Atualizar o histórico de versionamento:** Sempre que o documento for alterado o histórico deve ser atualizado, com quem atualizou, quando, e quais foram as alterações.
  - **Atualizar a Sessão de rastreamento:** O documento deve mostrar todos os documentos que possuem relação com ele, caso tenha, deve ser colocado em uma tabela o identificador do documento e fazer um link externo, o documento de requisitos por exemplo deve ter o identificador DRE junto a numeração corresponde ao documento como por exemplo 001 e assim por diante. Cada documento vai ter um identificador próprio e a numeração não deve ter relação com nenhum outro identificador.
- **Documento de requisitos:** Documento que contém os requisitos funcionais, não funcionais, regras de negócios, histórias de usuário, diagramas e descrições de casos de uso, será utilizado em quase todo o processo.
  - **Validação do documento:** Uma reunião entre os membros da equipe é feita para validar o documento, caso precise de mudanças, essas devem ser feitas o quanto antes.
  - **Desenvolvimento:** Com o documento de requisitos o mais completo possível, devem começar com as atividades de desenvolvimento nas sprints, mas sempre que encontrar algo que precise ser acrescentado ou mudado deve se atualizar o documento nas atividades anteriores.
    - **Identificar Funcionalidades:** Os desenvolvedores devem identificar todas as funcionalidades que precisam ser desenvolvidas, para então decidir entre os desenvolvedores qual ficará com quais funcionalidades durante a sprint.

- **Pesquisar por soluções mais otimizadas que possam contribuir para o desenvolvimento:** Antes de começar a desenvolver, seria bem recomendado fazer pesquisas, para ver o que já existe, talvez algoritmos altamente usados para resolver os problemas, ou que sejam problemas parecidos que possam ser aproveitados na aplicação para agilizar o desenvolvimento.
- **Desenvolver Funcionalidade:** Com a pesquisa feita, então o desenvolvedor responsável deve começar a desenvolver a funcionalidade com base em suas descrições, após a funcionalidade está completa então deve passar para os testes, durante o desenvolvimento é importante que as linhas de código sejam escritas de forma clara e intuitiva, além de deixar algumas partes do código bem comentadas.
- **Nova Funcionalidade:** É o arquivo ou classe com os códigos das novas funcionalidades.
- **Testes:** Os testes são uma parte importante, aqui algum membro da equipe que não necessariamente seja o desenvolvedor, deve realizar uma sequência de testes documentados para determinar se as funcionalidades estão fazendo o que foi especificado na documentação.
  - **Atualizar rastreabilidade do documento:** A seção do documento que faz referência aos documentos que o documento de teste faz referência, lembrando de diferenciar os identificadores por os respectivos documentos, como por exemplo documento de requisitos com o identificador DRE ou documento de teste DTE, e colocando a numeração correspondente ao documento e após deve linkar os documentos através de um link externo.
  - **Criar documentos de testes:** São os documentos mostra o passo a passo que o testador deve fazer para identificar se o sistema está fazendo o que deve fazer em todos os casos possíveis, além de documentar quem fez o teste, o resultado e caso tenha, uma solução ou sugestão, ou se a funcionalidade realmente faz o que deveria de acordo com as especificações.
  - **Identificar as funcionalidades desenvolvidas:** O testador deve verificar quais foram as funcionalidades desenvolvidas o quanto antes após seu desenvolvimento e fazer os testes.
  - **Efetuar todos os casos de teste:** Os casos de testes foram desenvolvidos com base no documento de requisitos, o testador deve realizar todos os casos para determinar se a funcionalidade faz ou não o que deve.
  - **Documentar todos os resultados:** Com os casos de teste feitos, o testador deve documentar todos os resultados encontrados no documento de teste.
  - **Passar funcionalidade para o Git:** Se caso a funcionalidade estiver funcionando corretamente ela deve ser colocada no repositório da empresa, para que todos os membros da equipe possam ter acesso e ter o controle sobre o desenvolvimento e marcar em todas as legendas das documentações, como completa, ou parcialmente desenvolvida, caso só uma parte dela seja desenvolvida.
  - **Voltar ao desenvolvimento:** O testador deve retornar a funcionalidade para lista de desenvolvimento, e marcar em todas as legendas nas documentações, que aquela funcionalidade ou falhou ou está incompleta.
- **Documento de teste:** Documento onde terá todos os casos de teste e os resultados dos testes realizados em cada funcionalidade, além de conter os nomes de quem realizou os testes e os resultados.

- **Reunião para controle de mudanças:** Sempre que alguma mudança deve ser feita uma reunião com a equipe e se possível com alguém que seja responsável pelo sistema, para discutir sobre as mudanças necessárias e saber os impactos que causará no sistema e se é viável ou não.
- **Criar relatório de mudanças:** Um membro da equipe deve criar um documento para identificar quem fez a mudança, por que fez a mudança e quando, sempre que algo for mudado deve ser adicionado a um relatório.
- **Relatório de mudanças:** Documento com todas as mudanças detalhadas, mostrando o que foi mudado, por quem, justificativa e quando a mudança foi feita, o documento sempre deve se manter atualizado, pois ele deve ser estudado por novos membros das equipes quando entrarem, para entender o motivo de determinadas escolhas de membros anteriores.
- **Reunião de encerramento:** Uma reunião geral, para mostrar o que foi feito durante a sprint, o que foi aprendido e o que pode ser melhorado para a próxima sprint.

Para esse processo é recomendado usar o trello, uma ferramenta de gestão de tarefas, assim a equipe saberá o que tem que fazer e o que cada um já faz, e o que estão fazendo, é bem simples de se usar e basta ter uma conta no google para usar.

Outra ferramenta são as disponibilizadas pelo google drive, como o docs, assim todos terão acesso aos documentos mais rapidamente, além de poder fazer apresentações e ter planilhas para o uso da equipe quando necessário.

Por fim o uso do Git para fazer a gestão do sistema, assim mantendo o controle da versão e que todos os membros da equipe tenham acesso aos mesmos códigos.

As entrevistas, questionários e observações ficam a critério dos desenvolvedores, pois eles variam facilmente conforme o sistema solicitado, porém ao final deste documento terá em Apêndice um exemplo de um documento de requisito completo usado por este processo, além do documento de teste e um modelo de relatório de mudanças.

## 5. Referências

CAVALCANTE, Mirlane Beserra. Proposta de um modelo baseado na combinação de metodologias ágeis de desenvolvimento de software. 2018. 115 f. Monografia (Graduação em Engenharia de Software) - Campus de Russas, Universidade Federal do Ceará, Russas, 2018.

MARTINS, Luiz Eduardo Galvão. Uma metodologia de elicitação de requisitos de software baseada na teoria da atividade. 2001. 170p. Tese (doutorado) - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação, Campinas, SP. Disponível em: <<http://www.repositorio.unicamp.br/handle/REPOSIP/260233>>. Acesso em: 17 ago. 2020.

MPS.BR - Melhoria de Processo do Software Brasileiro: Guia Geral MPS de Software, SofTex, 2020. Disponível em: <<https://softex.br/mpsbr/guias/#guia-sw>>. Acesso em: 17 ago. 2020.

O QUE É CMMI?. ISDBRASIL. Disponível em: <<http://www.isdbrasil.com.br/o-que-e-cmmi.php>>. Acesso em: 17 ago. 2020.

**6.** Apêndices**APÊNDICE 1 - Exemplo de documento de requisitos**

(Imagem representando a empresa)

# Documento de Requisitos

Projeto: [*Nome do Projeto*]

Russas - Ceará

26/03/2021

## Sumário

<b><u>1. INTRODUÇÃO</u></b>	<b><u>5</u></b>
<u>1.1. Visão geral do documento</u>	<u>5</u>
<u>1.2. Identificação dos requisitos</u>	<u>5</u>
<u>1.3. Prioridades dos requisitos</u>	<u>5</u>
<b><u>2. RASTREAMENTO DE DOCUMENTAÇÃO</u></b>	<b><u>5</u></b>
<b><u>3. DESCRIÇÃO GERAL DO SISTEMA</u></b>	<b><u>5</u></b>
<u>3.1. Sistema</u>	<u>5</u>
<u>3.2. Stakeholders</u>	<u>6</u>
<b><u>4. LISTAGEM DOS REQUISITOS</u></b>	<b><u>6</u></b>
<u>4.1 Regras de Negócio</u>	<u>6</u>
<u>4.2. Estórias de Usuários</u>	<u>6</u>
<u>Requisitos Funcionais</u>	<u>7</u>
<u>4.3. Requisitos Não-Funcionais</u>	<u>7</u>
<b><u>5. MODELAGEM DO SISTEMA</u></b>	<b><u>7</u></b>
<b><u>6. DESCRIÇÃO DE CASO DE USO</u></b>	<b><u>7</u></b>

## HISTÓRICO DE ALTERAÇÕES

*[Ao final de cada modificação no documento o histórico de versão deve ser atualizado, a mudança de versão nem sempre é necessária, apenas quando a alteração for grande o suficiente para isso, além disso manter a prática de manter as linhas alternadas]*

Data	Versão	Descrição	Autor	Alterações
------	--------	-----------	-------	------------

--	--	--	--	--

## GLOSSÁRIO

*[Todos os termos pouco conhecidos e que podem ser estranhos, abreviações e nomenclaturas devem ser colocadas na tabela juntamente com sua descrição como mostrado no exemplo a seguir]*

Segue abaixo a tabela de termos e suas respectivas descrições usadas no sistema:

Termo	Descrição
Requisitos funcionais	Características e ações que o sistema deve realizar para atender a necessidade do cliente.
Requisitos funcionais não	Descrevem atributos e restrições que o sistema deve ter.
<i>Stakeholders</i>	Pessoas ou organizações que serão afetadas pelo sistema e que, direta ou indiretamente, têm influência sobre os requisitos.
<i>Templates</i>	Modelos de documentos que indicam o conteúdo necessário. A Engenharia de <i>Software</i> utiliza-os com muita frequência para sugerir padrões.

## ● 1. INTRODUÇÃO

*[Esta seção do documento é destinada a introdução sobre o que se trata o documento de requisitos, essa sessão ainda será dividida em outras três subseções que serão mais específicas]*

### ○ 1.1. Visão geral do documento

*[Nesta subseção deve ser mostrado como o documento está dividido]*

### ○ 1.2. Identificação dos requisitos

*[Nesta subseção deve ser mostrado como será feito a identificação dos requisitos e explica cada uma delas]*

### ○ 1.3. Prioridades dos requisitos

*[Nesta subseção deve ser apresentado qual será a prioridade dos requisitos e qual a definição de cada prioridade]*

## ● 2. RASTREAMENTO DE DOCUMENTAÇÃO

*[Nesta seção deve ser colocado todos os documentos que serviram de referência direta para a criação deste documento, colocando o identificador do documento, junto com um link externo]*

## ● 3. DESCRIÇÃO GERAL DO SISTEMA

*[Esta seção será dividida em duas subseções]*

### ○ 3.1. Sistema

*[Nesta subseção deve ser feita uma descrição completa sobre o sistema que está sendo desenvolvido, informações como para o que serve, funcionalidades de forma geral, entre outras informações]*

### ○ 3.2. Stakeholders

*[Nesta subseção deve ser descrito quem são todos os Stakeholders do sistema]*

## ● 4. LISTAGEM DOS REQUISITOS

*[Esta seção será dividida em três subseções]*

#### ○ 4.1 Regras de Negócio

*[Nesta subseção as regras de negócio devem ser identificadas e descritas de forma legível]*

ID Regra de Negócio	Título	Descrição

#### ○ 4.2. Estórias de Usuários

*[Nesta subseção as estórias de usuário devem ser colocadas em quadros, especificados, identificados, claros e coesos]*

<b>ID:</b>
<p style="text-align: center;"><b>Título: “”</b></p> <p><b>Descrição:</b></p> <p><b>Prioridade:</b></p>
<b>Critérios de Aceitação:</b>

<b>Testes a Realizar:</b>

**Requisitos Funcionais**

*[Nesta subseção os requisitos funcionais devem ser identificados e descritos de forma legível, coesa e coerente]*

Identificador	Título	Prioridade	Descrição
---------------	--------	------------	-----------

--	--	--	--

**4.3. Requisitos Não-Funcionais**

*[Nesta subseção os requisitos não-funcionais devem ser identificados e descritos conforme as necessidades, legíveis, entendíveis e coerentes]*

Identificador	Título	Descrição
---------------	--------	-----------

--	--	--

--	--	--

- **5. MODELAGEM DO SISTEMA**

*[Nesta seção estarão todos os diagramas feitos para modelar o sistema, cada diagrama será uma subseção desta]*

- **6. DESCRIÇÃO DE CASO DE USO**

*[Nesta seção cada caso de uso deve ser descrito exatamente da forma que ele deve se comportar em todos os cenários possíveis]*

O nível de desenvolvimento do componente é exibido na legenda abaixo:



**Componente desenvolvido**



**Componente em desenvolvimento**



**Componente desenvolvido, mas apresentou falhas**



**Componente ainda não desenvolvido**



**Situação não identificada**

Lista de casos de uso descritos:

ID

Nome	
Ator Primário	
Sumário	
Pontos de Extensão	
Regras de Negócio	
Estórias de Usuário	
Requisitos Funcionais	
Pré-condições	
Pós-condições	
Fluxo Principal	
Fluxo Alternativo	
Fluxo de Exceção	

## APÊNDICE 2 - Exemplo de documento de Relatório de Mudanças

(imagem da empresa)

Relatório de Mudanças (*nome da empresa/sistema*)

Projeto: [*Nome do Projeto*]

Relatório de Mudanças

Nome	Data	Mudança	Motivo	Rastreamento de Documentação
Lucas Rafael	15/01/2021	A regra de negócio [RN013] foi removida.	A regra de negócio [RN008] já engloba a ocasião da regra de negócio [RN013].	[DR001]
Lucas Rafael	15/01/2021	O ator professor foi removido da documentação.	Pois foi se notado que a parte que condizia com o ator professor já não fazia mais parte do que o sistema deveria fazer.	[DR001], [DMU001], [DTE001]
Lucas Rafael	15/01/2021	O requisito funcional [RF007] foi alterado.	O requisito se mostrava ambíguo e repetia o que o requisito funcional [RF003] fazia, porém de forma confusa.	[DR001]

Apêndice 7 – Modelo de Engenharia de Requisitos Ágeis.

# Modelo de Engenharia de Requisitos Ágil

Modelo proposto por: Lucas Rafael Alves Oliveira

Março de 2021

**SUMÁRIO**

**Introdução**    **3**

**Objetivo**    **3**

**Modelo Gráfico**    **3**

**Descrição das Atividades**    **4**

**Referências**    **6**

**Apêndices**    **7**

## 1. Introdução

Pensando em como a engenharia de requisitos é importante e como algumas empresas têm dificuldade para fazer um bom processo de engenharia de requisitos, esse modelo foi criado, de forma a simplificar e agilizar o processo, visando principalmente a universidade em seus projetos e empresas sem nenhum membro experiente.

## 2. Objetivo

O processo tem como objetivo dar aos usuários do guia um processo de engenharia de requisitos simplificado, mas que ainda assim possua os quesitos de qualidade proposto pelo CMMI e MPS Br. Assim fornecendo agilidade, facilidade e qualidade.

## 3. Modelo Gráfico

Nesta sessão iremos ver o modelo gráfico do processo engenharia de requisitos, focando principalmente nas atividades realizadas em engenharia de requisitos visando a qualidade, como é mostrado nas Figuras 1 e 2.

Figura 1 - Modelo de Engenharia de Requisitos Ágil.

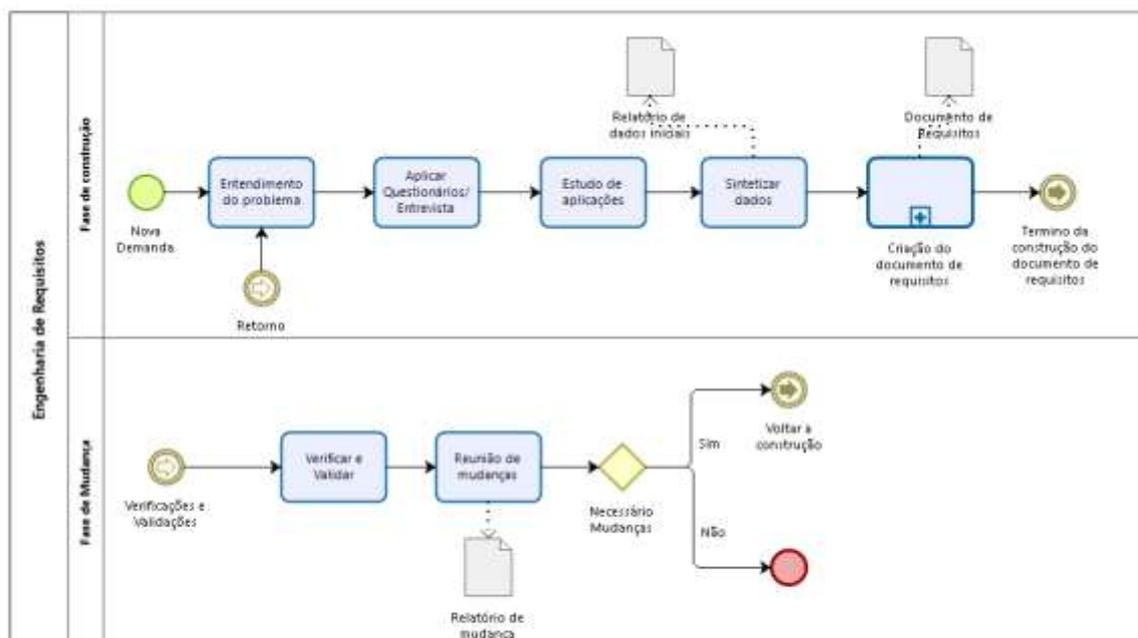
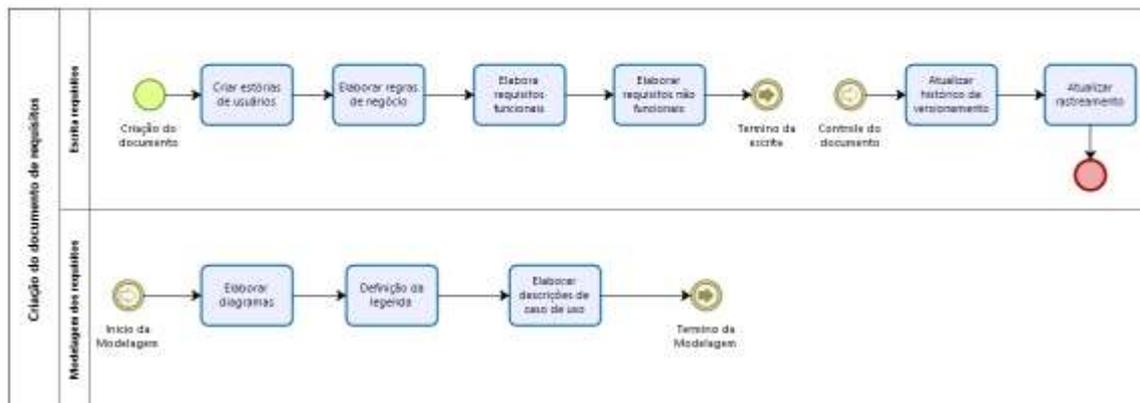


Figura 2 - Modelo de Engenharia de Requisitos Ágil Subprocesso de Criação do Documento de Requisitos.



Powered by  
**bizagi**  
Modeler

Fonte: Autor do Trabalho

#### 4. Descrição das Atividades

Nesta sessão veremos a descrição das atividades, eventos e artefatos mostrados no processo mostrado na Figura 1, será detalhado na mesma ordem de apresentação do modelo, mostrado em uma lista, além de incluir ferramentas que podem ser úteis no processo. Cada atividade, artefato ou evento mostrado anteriormente será listado e estarão destacados no texto em negrito.

- **Entendimento do problema:** Deve ser feitas pesquisas sobre o ambiente do sistema, buscar quem serão os possíveis usuários e a partir daí, elaborar, entrevistas e questionários, ou até mesmo pesquisar aplicações semelhantes.
- **Aplicar Questionários/Entrevista:** Entrevistar ou aplicar questionários nos possíveis usuários, ou até mesmo utilizar a observação para entender como o usuário faz o processo, com isso saberá quais são os principais requisitos a serem desenvolvidos.
- **Estudo de aplicações:** Entender como as aplicações semelhantes se comportam, o que eles já fazem, o que não faz, o que o usuário quer ver, o que ele quer que seja melhor, pode ser feito um questionário diretamente voltado para essa pesquisa de mercado com usuários que já usam essas aplicações, se for possível.
- **Sintetizar dados:** Com os dados colhidos, de questionários, entrevistas, observações e pesquisa, devem ser passados para relatórios e planilha, para conseguir ver amplamente a realidade dos dados, com isso poderá ter a noção do que os usuários responderam nos questionários, um apanhado das informações mais importantes, das entrevistas, observações e pesquisas.
- **Relatório de dados iniciais:** Esse documento conterá as informações organizadas, que foram coletadas nas entrevistas, questionários, observações e pesquisas, ele servirá de base para o documento de requisitos.

- **Criação do documento de requisitos:** Preparar o documento para receber as informações, adicionar uma introdução onde vai conter as informações básicas do documento, um glossário, onde serão colocadas as informações, como abreviações, jargões, termos desconhecidos.
  - **Criar estórias de usuários:** Com base no documento de dados iniciais, as estórias de usuário devem ser criadas, vindo do ponto de vista do próprio usuário, com isso já vai ser possível ter uma noção maior de como o sistema deve funcionar, dando uma visão mais ampla.
  - **Elaborar regras de negócio:** Com base nas informações já encontradas, coloque as regras de negócio em uma tabela, com o identificador, nome e a descrição, as regras de negócio vão dizer algumas regras que o sistema deve seguir.
  - **Elabora requisitos funcionais:** Com base no que já se sabe sobre o sistema, os requisitos funcionais devem ser criados ou modificados com os seus identificadores, nome, descrição e prioridade, eles são tudo que o sistema de fato deve fazer.
  - **Elaborar requisitos não funcionais:** Com base no que já se sabe sobre o sistema os requisitos não funcionais devem ser criados ou modificados, com seus identificadores, nomes e descrição, os requisitos não funcionais, são restrições para o sistema, mas que não afeta tanto assim seu comportamento e sim sua eficiência, qualidade e construção.
  - **Elaborar diagramas:** Os diagramas devem ser construídos conforme sua necessidade, iniciando com o diagrama de caso de uso, pois ele é um dos mais úteis para demonstrar como o sistema vai se comunicar com os usuários e as funcionalidades, cada caso de uso deve possuir seu identificador, para mapear na descrição de caso de uso, conforme novos diagramas forem sendo necessários podem ser criados.
  - **Definição da legenda:** Uma legenda deve ser definida para identificar o que foi feito, o que tem para fazer, o que precisa ser revisado, o que está pronto ou incompleto.
  - **Elaborar descrições de caso de uso:** Para cada caso de uso, deve ser criado uma descrição, com o identificador, ator responsável, condições de entrada e saída, e como a funcionalidade deve se comportar, além de ter que descrever os casos excepcionais que podem ocorrer, todas as descrições devem ser detalhadas, pois as funcionalidades serão construídas das descrições e não pode haver confusão, ambiguidade, incompletude ou coesão, devem estar o mais legível possível.
  - **Atualizar histórico de versionamento:** O histórico deve ser atualizado sempre que o documento de requisitos for alterado, colocando o que foi modificado ou incrementado, a data, a versão, e quem fez.
  - **Atualizar rastreamento:** O rastreamento do documento deve ser feito sempre que for necessário, uma vez que novos documentos podem surgir e é de extrema importância que todos saibam de onde as informações do documento de requisitos vieram.
- **Documento de requisitos:** Documento que servirá de base para o desenvolvimento do sistema, contém as regras de negócio, requisitos funcionais e não funcionais, diagramas, descrição dos casos de uso, além de ser importante para manter o sistema.
- **Verificar e Validar:** Pode ser feito com questionários e entrevistas com usuários, feitos com base no documento, para confirmar se as funcionalidades estão corretas.

- **Reunião de mudanças:** Devem ocorrer quando é necessário fazer mudanças na documentação, para avaliar o que deve ser feito, os impactos das mudanças, os custos e a dificuldade de mudar.
- **Relatório de mudança:** Documento que deve ser feito com as mudanças, para que elas fiquem registradas para que possa ser acessado quando for necessário ter as informações que foram mudadas, o documento deve conter o responsável, a data, a mudança, o porquê da mudança e o rastreamento da mudança, onde a mudança foi feita.

Obs: Para melhor o desempenho do trabalho em equipe é recomendado o uso do google drive compartilhado entre a equipe, assim todos vão poder ter maior acesso a informação simultaneamente.

Para o acompanhamento das atividades o uso do trello é bastante recomendado, assim todos da equipe já vão poder ter acesso mais facilmente o que cada uma faz.

## 5. Referências

MARTINS, Luiz Eduardo Galvão. Uma metodologia de elicitação de requisitos de software baseada na teoria da atividade. 2001. 170p. Tese (doutorado) - Universidade Estadual de Campinas, Faculdade de Engenharia Eletrica e de Computação, Campinas, SP. Disponível em: <<http://www.repositorio.unicamp.br/handle/REPOSIP/260233>>. Acesso em: 17 ago. 2020.

MPS.BR - Melhoria de Processo do Software Brasileiro: Guia Geral MPS de Software, SofTex, 2020. Disponível em: <<https://softex.br/mpsbr/guias/#guia-sw>>. Acesso em: 17 ago. 2020.

O QUE É CMMI?. ISDBRASIL. Disponível em: <<http://www.isdbrasil.com.br/o-que-e-cmmi.php>>. Acesso em: 17 ago. 2020.

## 6. Apêndices

### APÊNDICE 1 - Exemplo de documento de requisitos

(Imagem representando a empresa)

## Documento de Requisitos

Projeto: [*Nome do Projeto*]

Russas - Ceará

26/03/2021

## Sumário

<b><u>1. INTRODUÇÃO</u></b>	<b><u>5</u></b>
<u>1.1. Visão geral do documento</u>	<u>5</u>
<u>1.2. Identificação dos requisitos</u>	<u>5</u>
<u>1.3. Prioridades dos requisitos</u>	<u>5</u>
<b><u>2. RASTREAMENTO DE DOCUMENTAÇÃO</u></b>	<b><u>5</u></b>
<b><u>3. DESCRIÇÃO GERAL DO SISTEMA</u></b>	<b><u>5</u></b>
<u>3.1. Sistema</u>	<u>5</u>
<u>3.2. Stakeholders</u>	<u>6</u>
<b><u>4. LISTAGEM DOS REQUISITOS</u></b>	<b><u>6</u></b>
<u>4.1 Regras de Negócio</u>	<u>6</u>
<u>4.2. Estórias de Usuários</u>	<u>6</u>
<u>Requisitos Funcionais</u>	<u>7</u>
<u>4.3. Requisitos Não-Funcionais</u>	<u>7</u>
<b><u>5. MODELAGEM DO SISTEMA</u></b>	<b><u>7</u></b>
<b><u>6. DESCRIÇÃO DE CASO DE USO</u></b>	<b><u>7</u></b>

## HISTÓRICO DE ALTERAÇÕES

*[Ao final de cada modificação no documento o histórico de versão deve ser atualizado, a mudança de versão nem sempre é necessária, apenas quando a alteração for grande o suficiente para isso, além disso manter a prática de manter as linhas alternadas]*

Data	Versão	Descrição	Autor	Alterações
------	--------	-----------	-------	------------

--	--	--	--	--

## GLOSSÁRIO

*[Todos os termos pouco conhecidos e que podem ser estranhos, abreviações e nomenclaturas devem ser colocadas na tabela juntamente com sua descrição como mostrado no exemplo a seguir]*

Segue abaixo a tabela de termos e suas respectivas descrições usadas no sistema:

Termo	Descrição
Requisitos funcionais	Características e ações que o sistema deve realizar para atender a necessidade do cliente.
Requisitos funcionais não	Descrevem atributos e restrições que o sistema deve ter.
<i>Stakeholders</i>	Pessoas ou organizações que serão afetadas pelo sistema e que, direta ou indiretamente, têm influência sobre os requisitos.
<i>Templates</i>	Modelos de documentos que indicam o conteúdo necessário. A Engenharia de <i>Software</i> utiliza-os com muita frequência para sugerir padrões.

## ● 1. INTRODUÇÃO

*[Esta seção do documento é destinada a introdução sobre o que se trata o documento de requisitos, essa sessão ainda será dividida em outras três subseções que serão mais específicas]*

### ○ 1.1. Visão geral do documento

*[Nesta subseção deve ser mostrado como o documento está dividido]*

### ○ 1.2. Identificação dos requisitos

*[Nesta subseção deve ser mostrado como será feito a identificação dos requisitos e explica cada uma delas]*

### ○ 1.3. Prioridades dos requisitos

*[Nesta subseção deve ser apresentado qual será a prioridade dos requisitos e qual a definição de cada prioridade]*

## ● 2. RASTREAMENTO DE DOCUMENTAÇÃO

*[Nesta seção deve ser colocado todos os documentos que serviram de referência direta para a criação deste documento, colocando o identificador do documento, junto com um link externo]*

## ● 3. DESCRIÇÃO GERAL DO SISTEMA

*[Esta seção será dividida em duas subseções]*

### ○ 3.1. Sistema

*[Nesta subseção deve ser feita uma descrição completa sobre o sistema que está sendo desenvolvido, informações como para o que serve, funcionalidades de forma geral, entre outras informações]*

### ○ 3.2. Stakeholders

*[Nesta subseção deve ser descrito quem são todos os Stakeholders do sistema]*

## ● 4. LISTAGEM DOS REQUISITOS

*[Esta seção será dividida em três subseções]*

#### ○ 4.1 Regras de Negócio

*[Nesta subseção as regras de negócio devem ser identificadas e descritas de forma legível]*

ID Regra de Negócio	Título	Descrição

#### ○ 4.2. Estórias de Usuários

*[Nesta subseção as estórias de usuário devem ser colocadas em quadros, especificados, identificados, claros e coesos]*

<b>ID:</b>
<p style="text-align: center;"><b>Título: “”</b></p> <p><b>Descrição:</b></p> <p><b>Prioridade:</b></p>
<b>Critérios de Aceitação:</b>

<b>Testes a Realizar:</b>

**Requisitos Funcionais**

*[Nesta subseção os requisitos funcionais devem ser identificados e descritos de forma legível, coesa e coerente]*

Identificador	Título	Prioridade	Descrição
---------------	--------	------------	-----------

--	--	--	--

**4.3. Requisitos Não-Funcionais**

*[Nesta subseção os requisitos não-funcionais devem ser identificados e descritos conforme as necessidades, legíveis, entendíveis e coerentes]*

Identificador	Título	Descrição
---------------	--------	-----------

--	--	--

--	--	--

- **5. MODELAGEM DO SISTEMA**

*[Nesta seção estarão todos os diagramas feitos para modelar o sistema, cada diagrama será uma subseção desta]*

- **6. DESCRIÇÃO DE CASO DE USO**

*[Nesta seção cada caso de uso deve ser descrito exatamente da forma que ele deve se comportar em todos os cenários possíveis]*

O nível de desenvolvimento do componente é exibido na legenda abaixo:



**Componente desenvolvido**



**Componente em desenvolvimento**



**Componente desenvolvido, mas apresentou falhas**



**Componente ainda não desenvolvido**



**Situação não identificada**

Lista de casos de uso descritos:

ID
----

Nome	
Ator Primário	
Sumário	
Pontos de Extensão	
Regras de Negócio	
Estórias de Usuário	
Requisitos Funcionais	
Pré-condições	
Pós-condições	
Fluxo Principal	
Fluxo Alternativo	
Fluxo de Exceção	

**APÊNDICE 2 - Exemplo de documento de Relatório de Mudanças**

<b>Nome</b>	<b>Data</b>	<b>Mudança</b>	<b>Motivo</b>	<b>Rastreamento de Documentação</b>
Lucas Rafael	15/01/2021	A regra de negócio [RN013] foi removida.	A regra de negócio [RN008] já engloba a ocasião da regra de negócio [RN013].	[DR001]
Lucas Rafael	15/01/2021	O ator professor foi removido da documentação.	Pois foi se notado que a parte que condizia com o ator professor já não fazia mais parte do que o sistema deveria fazer.	[DR001], [DMU001], [DTE001]
Lucas Rafael	15/01/2021	O requisito funcional [RF007] foi alterado.	O requisito se mostrava ambíguo e repetia o que o requisito funcional [RF003] fazia, porém de forma confusa.	[DR001]