
Modelo de Processo S.C.U.

*Felipe Issamu de Melo Kamimura, Isabelle Ichikawa Yagi, Lucas
Freitas Costa, Lucas Tarumoto, Nathiely Moraes Macedo, Raniel
Carlos Bispo dos Santos*

Universidade Tecnológica Federal do Paraná - Campus Cornélio Procópio

Este documento descreve um modelo de processo de desenvolvimento de
Software S.C.U.

26 de setembro de 2018

Lista de figuras

1	Modelo do Processo	6
2	Diagrama BPMN: Levantamento de Requisitos	7
3	Diagrama BPMN: Projeto e Implementação	8
4	Diagrama BPMN: Verificação e validação	9
5	Diagrama BPMN: Implantação	10
6	Diagrama BPMN: Manutenção e Evolução	10
7	Template Relatório Geral	24
8	Template de Relatório de Instalação	25
9	Template ata de reunião de finalização de fase	26
10	Template ata de reunião de incialização do Projeto	27
11	Template ata de reunião de iniciação de Sprint	28

Lista de tabelas

1	Tabela com descrição dos papeis	11
2	Enumeração e descrição das atividades vinculados com os papeis	12
3	Cronograma com as atribuições aos integrantes	16
4	Tabela de Checklist de Artefatos	18
5	Tabela de Checklist de Atividades	21

Sumário

1	Introdução	4
2	Processo	4
2.1	Papeis	11
2.2	Atividades	12
3	Execução do projeto	16
3.1	Backlog e sprints	16
3.2	Estado atual	18
4	Referências bibliográficas	23
5	Anexos	23

1 Introdução

O presente documento descreve um processo de produção de software construído com base no Scrum [1], no Cascata [2] e no Processo Unificado [3], o SCU, que busca unir conceitos de metodologia ágil com metodologia clássica, o objetivo é um processo simples de coordenar e bastante interativo, onde a comunicação entre os envolvidos e o comprometimento com cada fase do processo são evidenciados ao longo do desenvolvimento.

Assim os integrantes que realizarão o projeto serão estes apresentados a seguir, cada um com seu respectivo papel:

- Felipe Kamimura - Integrante do Scrum Team
- Isabelle Ichikawa - Scrum Master
- Lucas Freitas - Integrante do Scrum Team
- Lucas Tarumoto - Integrante do Scrum Team
- Nathiely Moraes - Integrante do Scrum Team
- Raniel Santos - Integrante do Scrum Team

O Product Owner deve ser o cliente do projeto que será desenvolvido utilizando o processo SCU. O endereço do repositório do projeto é:

<https://github.com/FelipeKamimura/GerenciaDeConfig-SCU>.

2 Processo

O processo foi implementado com elementos do Scrum [1], do Processo Unificado [3] e do modelo Cascata [2]. As fases constantes no cascata [2] foram mantidas no SCU, sendo consideradas sprints, que possuem reuniões no início de cada fase, onde são discutidos os objetivos da sprint, as dificuldades encontradas no desenvolvimento da sprint anterior e os avanços alcançados (conforme templates de reuniões).

Para garantir que o processo tenha qualidade, assim como o scrum, o processo proposto segue as diretrizes do modelo de qualidade CMMI, pois se trata de uma metodologia ágil. Este modelo requer padronização nos processos, sendo assim, todas as fases do SCU são definidas nesse documento, além dos relatórios e reuniões que possuem um padrão de documentação aqui disponibilizados. O planejamento dos prazos também são detalhados neste documento, a fim de tornar o processo gerenciado. O processo atende às requisições de definido, sendo descrito detalhadamente todos os seus procedimentos no presente documento.

Do processo unificado foram extraídos os artefatos e os papéis [3], entretanto os papéis do scrum [1] também fazem parte do processo, sendo eles o Scrum Master, Product Owner e Scrum Team [1] (os papéis são detalhados no item 2.1 deste documento. Além das sprints tradicionais do Scrum [1], no SCU existem as subsprints, que são conjuntos de atividades, podendo ser feitas em paralelo ou não, com determinado prazo de entrega. A diferenciação em sprints e subsprints se deve às reuniões, pois as sprints dependem das reuniões realizadas antes de sua iniciação, já as subsprints não possuem reuniões formais. A retroatividade do modelo cascata também é mantida, dessa forma, caso haja mudanças nos requisitos, por exemplo, pode-se voltar pra primeira fase e realizar as alterações.

A primeira coisa a ser realizada no processo é a reunião de iniciação, o objetivo desta reunião é envolver todos os stakeholders no processo, é o momento em que o Scrum Master mostra o contexto do projeto para sua equipe e orienta os membros em relação às suas atividades, em seguida as subsprints são definidas e o prazo de entrega estipulado.

Após realizar a reunião de iniciação, dá-se o início das Sprints, que são as fases do modelo cascata. A primeira fase se trata do Levantamento de Requisitos [2], que é formado por cinco subsprints com um total de duração de 6 semanas. O longo prazo dessa fase se deve a complexidade das subsprints, os requisitos são fundamentais para o sucesso do projeto, portanto o foco nessa parte do processo é muito importante. As subsprints e suas atividades podem ser visualizadas no item 3 deste documento, os artefatos gerados estão no item 3.2.

Ao final das 6 semanas, é realizada a reunião de iniciação sprint, a organização que adota o processo define como serão realizadas as reuniões, não existe nenhuma obrigatoriedade quanto à duração ou permanecer em pé como acontece no scrum, o único requisito é que seja documentada conforme o template disponibilizado.

Com a reunião concluída e documentada, é iniciada a segunda Sprint, que é a fase de Projeto e Implementação do cascata [2], é importante ressaltar que os integrantes da equipe (definido pelo Scrum Team) podem assumir diferentes funções ao longo do processo. Esta fase tem a duração de 7 semanas com 3 subsprints, sendo 4 semanas reservadas para a subsprint de implementação, que consiste na codificação e descrição do código do sistema. Ao final da sprint, novamente é realizada uma reunião de iniciação da sprint.

A terceira sprint do SCU é a fase de Verificação e Validação, com a duração de 2 semanas, essa fase é rápida, porém deve ser realizada com muita atenção, é nela que serão identificados os problemas decorrentes no sistema por meio de testes que devem ser executados com base num conjunto de casos de teste definidos pela organização/Scrum master juntamente com a equipe, que é o plano de teste. Esta fase pode identificar mudanças nos requisitos, por isso é importante realizar uma avaliação dos testes, que ajuda a decidir se essas mudanças são realmente plausíveis, pois nada impede que um teste seja conduzido incorretamente levando a resultados incoerentes com o objetivo da aplicação. Ao esgotar o prazo da sprint e concluir os testes, é o momento de realizar a reunião e dar início a quarta sprint.

A próxima sprint consiste na fase de Implantação do cascata [2], no SCU é estabelecida num período de 2 semanas, com 4 subsprints definidas. Nessa parte do processo o sistema é disponibilizado para os usuários, portanto deve ser entregue um manual e os treinamentos devem ser realizados para utilização do software, além disso, a instalação deve ser documentada conforme o relatório de instalação disponibilizado, isto é importante para controlar a distribuição do sistema. A utilização do sistema pelos usuários pode acarretar em novas mudanças ou aparecimento de bugs, essas informações devem constar no relatório de implantação ou ser discutidas e documentadas na reunião para iniciação da próxima Sprint.

A última sprint é a fase de Manutenção e Evolução do cascata [2], essa fase possui 5 subsprints, a subsprint chamada “gerência de alteração” tem a duração de 2 semanas, é nela que são corrigidos bugs percebidos durante a utilização do sistema e adotadas rotinas para manutenção e testes constantes do produto. Além disso, essa sprint controla as versões do sistema, as possíveis alterações e informações relacionadas ao progresso e resultados do software, as subsprints responsáveis por esses dados não possuem duração, pois acompanham o desenvolvimento e a evolução do produto.

O esquema do processo pode ser visualizado na Figura 1.

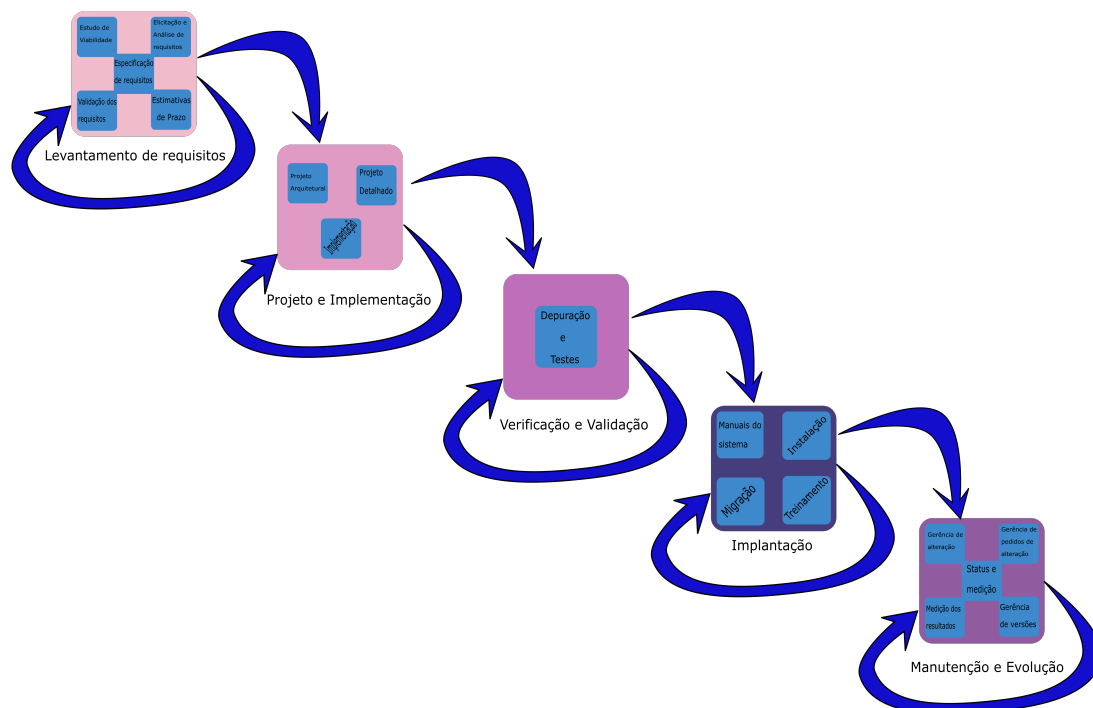


Figura 1: *Modelo do Processo*

O BPMN é o Modelo e Notação de Processos de Negócio que permite visualizar o gerenciamento dos processos de negócio do projeto que será desenvolvido. Os modelos foram divididos por fases, como podemos observar nas imagens abaixo.

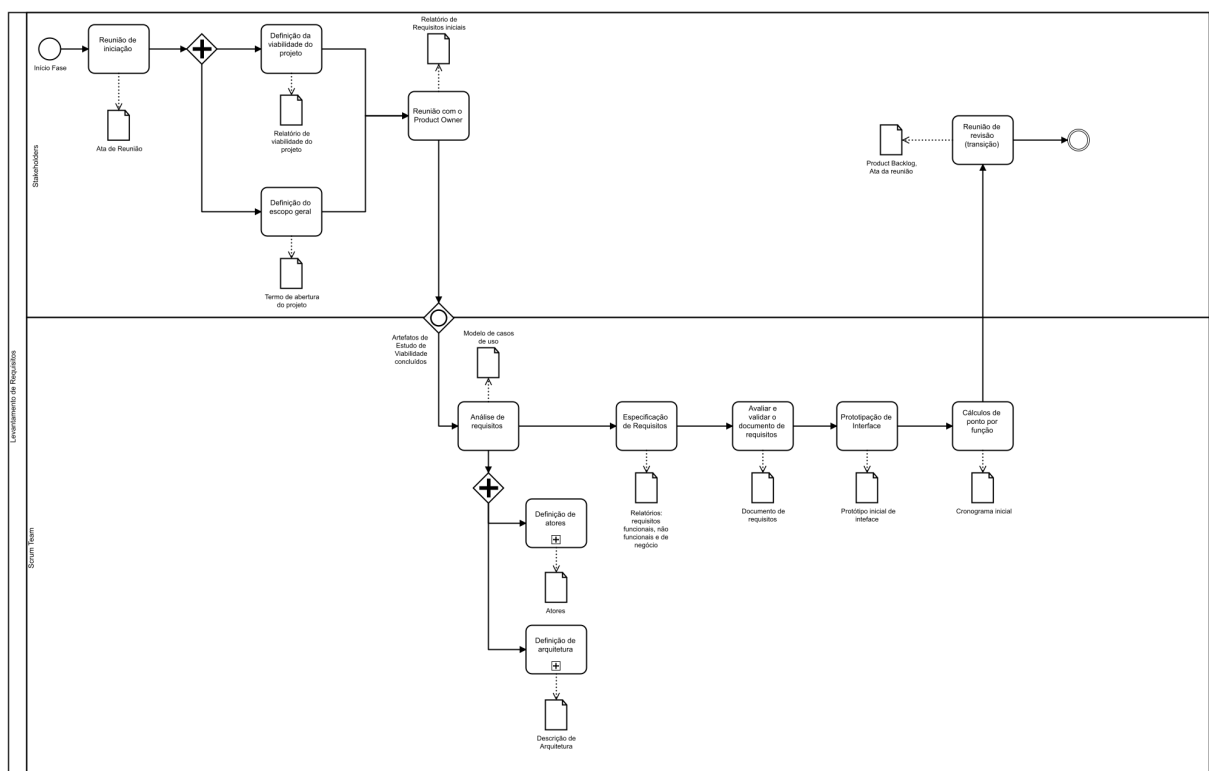


Figura 2: *Diagrama BPMN: Levantamento de Requisitos*

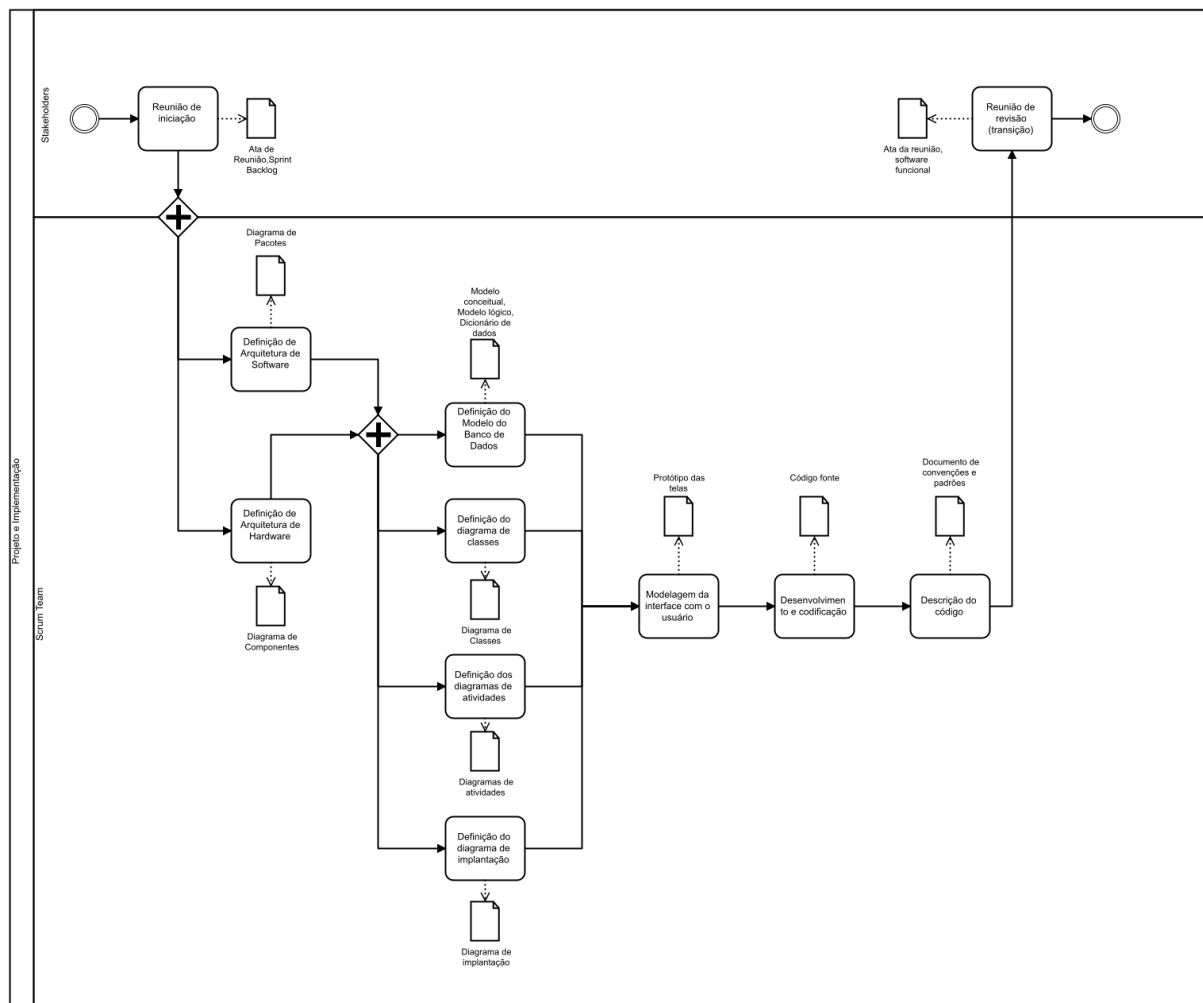


Figura 3: Diagrama BPMN: Projeto e Implementação

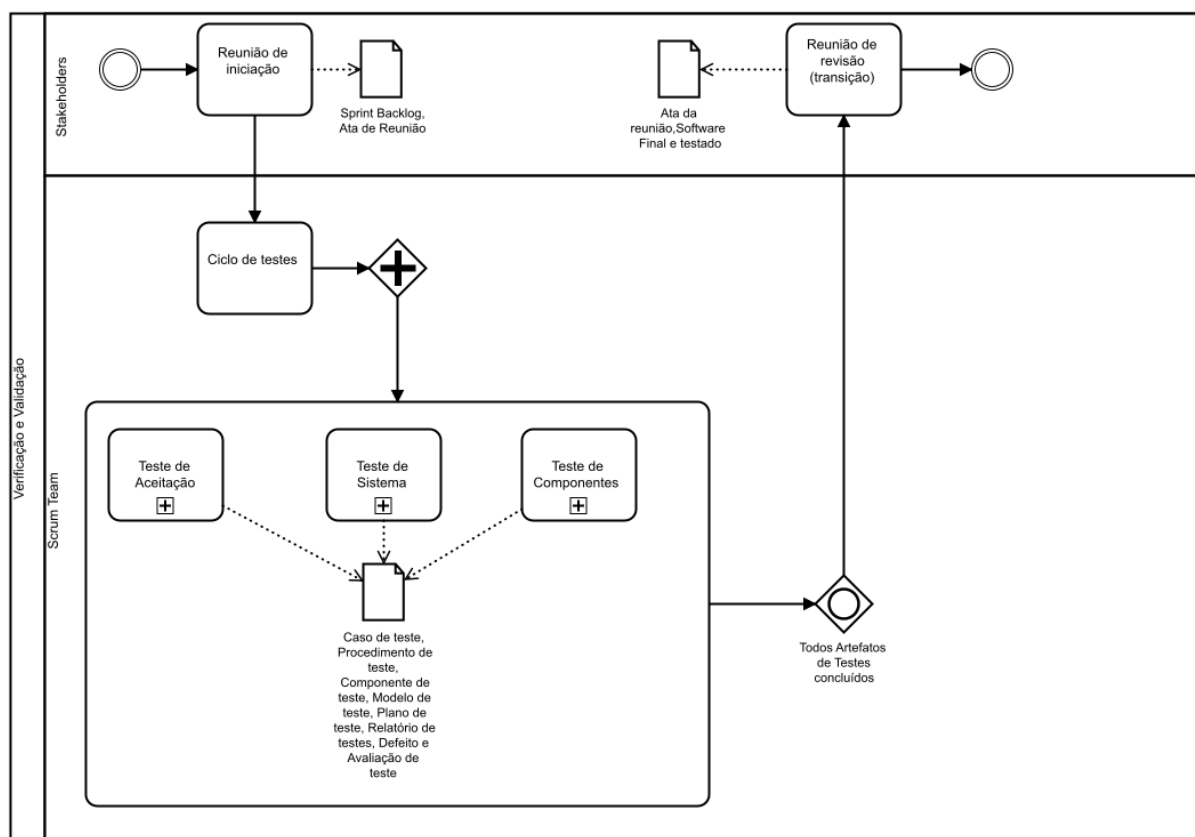


Figura 4: Diagrama BPMN: Verificação e validação

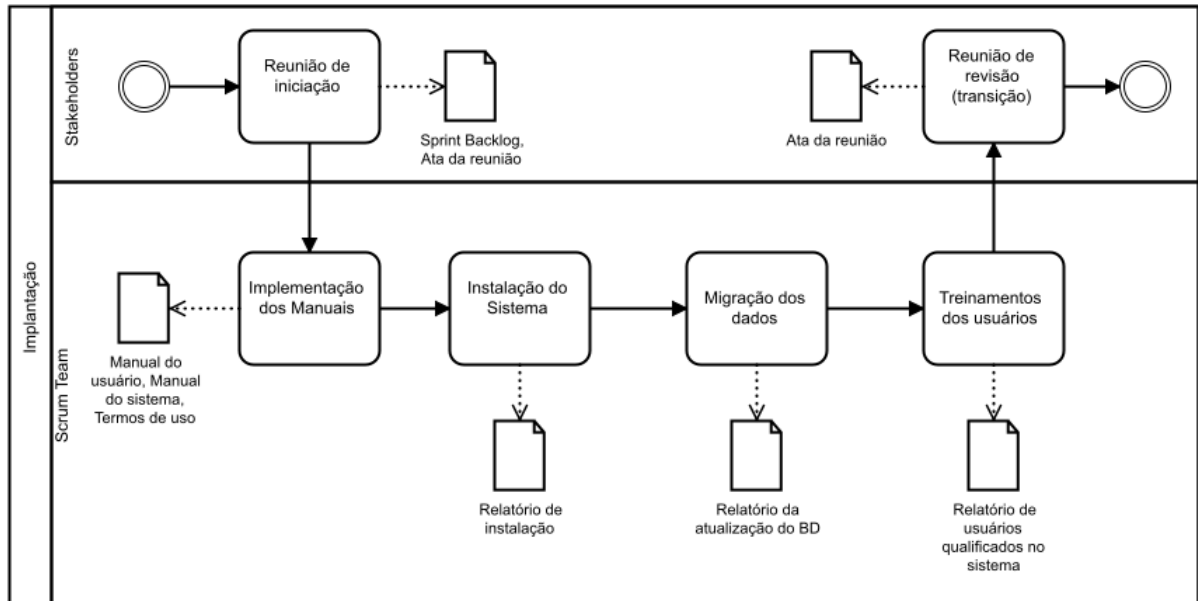


Figura 5: Diagrama BPMN: Implantação

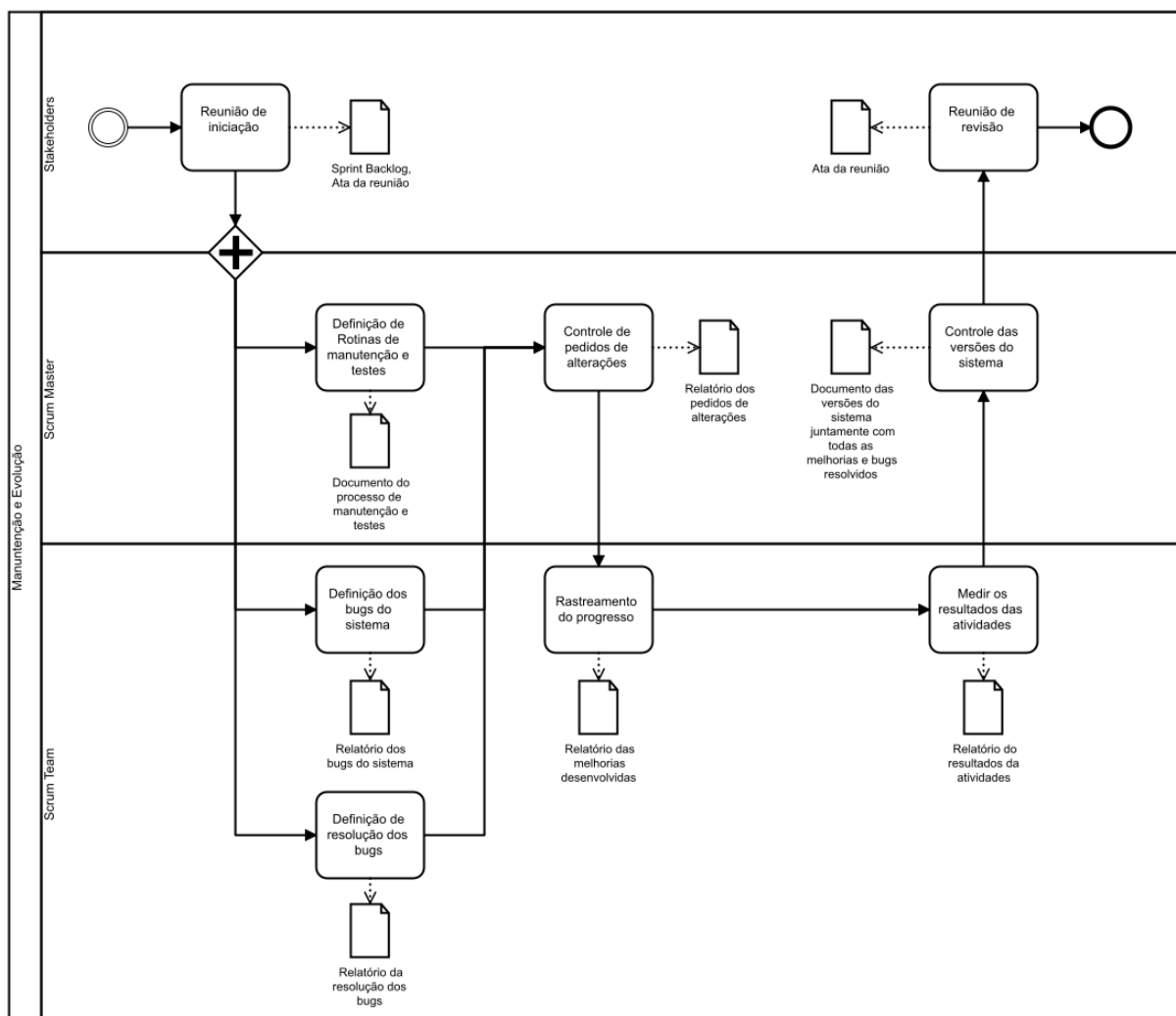


Figura 6: Diagrama BPMN: Manutenção e Evolução

2.1 Papéis

No projeto foram utilizados os papéis da metodologia Scrum, onde são divididos em Product Owner, Scrum Master e Scrum Team. O Product Owner seria aquele que possui conhecimento dos requisitos do projeto, no caso o sujeito que deseja o sistema. Já o Scrum Master pode ser exercido por qualquer pessoa da equipe e é responsável por garantir que processo seja seguido e prever possíveis entraves que possam acarretar em atrasos no cronograma. No papel de Scrum Team os integrantes são flexíveis e podem desenvolver em todas áreas do projeto. Para melhor especificação a Tabela 1 descreve cada função existente que será necessária no projeto.//

Tabela 1: *Tabela com descrição dos papéis*

Função	Descrição	Código
Gerente de projeto	Responsável por alocar recursos, garantir qualidade e integridade do sistema e definir prioridades	F01
Arquiteto de software	Responsável por definir interações do sistema, interfaces e por buscar soluções para problemas encontrados durante a construção do sistema	F02
Analista	Mão de obra principal, responsável pela escrita dos códigos e construção do sistema	F03
Engenheiro de Teste	Responsável por planejar os testes do sistema e avaliar os resultados posteriormente	F04
Tester	Responsável por efetuar os testes do sistema	F05
Arquiteto de Banco de Dados	Define o tipo de banco de dados a ser implementado, bem como sua arquitetura	F06
Designer	Desenvolve a interface gráfica de acordo com os modelos definidos pelo projetista de interfaces.	F07
Projetista de Interfaces	Determina a forma visual dos elementos da interface gráfica do sistema. (Desenvolve protótipos de tela)	F08
Analista de Requisitos	Responsável por mapear os requisitos ditos pelos stakeholders, mapear outros requisitos que irão aperfeiçoar o processo de desenvolvimento.	F09

2.2 Atividades

A Tabela 2 descreve cada uma das atividades do processo, relacionando com o respectivo papel responsável por sua execução.

Tabela 2: *Enumeração e descrição das atividades vinculados com os papéis*

	ATIVIDADE	DESCRIÇÃO DE ATIVIDADES	PAPÉIS
1.1	Reunião de iniciação	Reunião para marcar os encontros com os stakeholders para que todos saibam o que é esperado dessa primeira fase, serão definidos os tempos para a execução das sprints e como podem ser realizadas	Stakeholders
1.2	Definição da viabilidade do projeto	Verificar que com o orçamento proposto e tecnologias prontas para serem utilizadas são suficientes para a execução do projeto	Stakeholders
1.3	Definição de escopo geral	Definir linguagem, limites e restrições, que problemas que devem ser resolvidos e como serão resolvidos	Stakeholders
1.4	Reunião com o Product Owner	Reunião para verificar se o cliente está de acordo com o escopo definido	Stakeholders
1.5	Análise de requisitos	Onde serão realizados os modelos de casos de uso para auxiliar na análise dos requisitos necessários e verificar as atividades e funcionalidades que precisam estar presentes no software a ser desenvolvido	Scrum Team
1.6	Definição dos atores	A partir dos casos de usos são listados e verificados os atores envolvidos no sistema	Scrum Team
1.7	Definição da arquitetura	São especificados os casos de uso gerais sobre as funcionalidades essenciais do sistema	Scrum Team
1.8	Especificação dos requisitos	Com as funcionalidades definidas os requisitos funcionais e não funcionais podem ser mais especificados com o fim de validar a necessidade, se não estão ambíguos, e se suprem as demandas	Scrum Team
1.9	Avaliar e validar o documento de requisitos	A partir dos relatórios dos requisitos, os requisitos são analisados em busca de verificar se possuem rastreabilidade, se estão íntegros, correlacionados e se o documento de requisitos são	Stakeholders

Tabela 2: Enumeração e descrição das atividades vinculados com os papéis (continuação)

	ATIVIDADE	DESCRIÇÃO DE ATIVIDADES	PAPÉIS
1.10	Prototipação de interface	Com o uso de um design sprint será feito o estudo de um planejamento sobre a interface gráfica do sistema, para certificar que todos os requisitos estarão sendo preenchidos	Scrum Team
1.11	Cálculo de pontos por função	Com o uso dos caso de uso, será realizado um cronograma inicial para se obter uma estimativa de prazo de execução do projeto	Scrum Team
1.12	Reunião de revisão (transição)	Reunião de revisão de fase onde é verificado se o planejado foi executado, se há necessidade de mudança na execuções das sprints antes do inicio da próxima fase. É também elaborado o product backlog a partir de tudo que foi elicitado e também definido suas prioridades	Stakeholders
2.1	Reunião de iniciação	Definido o sprint backlog para que os parâmetros das sprints sejam passados para o scrum team	Stakeholders
2.2	Definição da arquitetura de hardware	São definidos todos os componentes (entregas) referentes ao projeto	Scrum Team
2.3	Definição da arquitetura de software	Através do diagrama de pacote são definidos como os pacotes serão estruturados dentro da arquitetura do sistema	Scrum Team
2.4	Definição do modelo de banco de dados	Nessa atividade é estudado os itens que precisam estar no banco de dados, se será relacional ou não relacional, atributos, entidades, chaves e relacionamentos	Scrum Team
2.5	Definição do diagrama de classes	Desenvolvimento dos diagramas de classes para que se tenha a visão geral e como as classes serão relacionadas umas com as outras, seus métodos e organização em relação aos objetos, além de separar elementos de design dos elementos de sistema	Scrum Team
2.6	Definição dos diagramas de atividades	Definir o fluxo do funcionamento e controle das atividades realizadas pelo sistema e como se interligam	Scrum Team

Tabela 2: *Enumeração e descrição das atividades vinculados com os papéis (continuação)*

	ATIVIDADE	DESCRIÇÃO DE ATIVIDADES	PAPÉIS
2.7	Definição do diagrama de implantação	Desenvolvimento do Diagrama de Implantação que especifica os requisitos mínimos para o funcionamento do sistema tais como as bibliotecas e softwares necessários, assim como as definições de hardware	Scrum Team
2.8	Modelagem da interface com o usuário	Diagramação para planejamento da tela e disposição dos itens na interface e estudo de como o usuário pode ter uma boa experiência de utilização. Usa como base o protótipo inicial porém na aplicação de heurísticas e experiência de usuário'	Scrum Team
2.9	Desenvolvimento e codificação	Codificação	Scrum Team
2.10	Descrição do código	São documentadas as formas e padrões adotados na codificação, além de descrever determinados blocos de código-fonte	Scrum Team
2.11		Reunião de revisão de fase onde é verificado se o planejado foi executado, se há necessidade de mudança na execuções das sprints antes do inicio da proxima fase. Se o software atende ao escopo especificado	Stakeholders
3.1		Reunião de inicio de fase, onde será definido como serão realizados os testes	Stakeholders
3.2	Ciclo de testes		Realização dos testes e correção no software, se necessário. Os testes devem ser relatados
	Scrum Team		
3.3	Teste de Componentes		Scrum Team
3.4	Teste de Sistema		Scrum Team
3.5	Teste de Aceitação		Scrum Team
3.6	Reunião de revisão (transição)	Reunião de finalização do software, realizada com os os stakeholders a fim de validar o sistema de acordo com o que era esperado	Stakeholders

Tabela 2: Enumeração e descrição das atividades vinculados com os papéis (continuação)

	ATIVIDADE	DESCRIÇÃO DE ATIVIDADES	PAPÉIS
4.1	Reunião de iniciação	Reunião de início de fase, onde é passado aos stakeholders como serão os procedimentos finais de desenvolvimento e como deve ser documentado as atualizações do sistema	Stakeholders
4.2	Implementação dos Manuais	Desenvolvimento dos Manuais do usuário e sistema para treinamento ou ajuda adicional para usuários, além do desenvolvimento dos Termos de Uso para o software	Scrum Team
4.3	Instalação do sistema	Instalação do sistema no ambiente requisitado pelo Product Owner seguindo as especificações	Scrum Team
4.4	Migração dos dados	Transferência de dados antigos para o banco de dados arquitetado resumindo no mesmo atualizado	Scrum Team
4.5	Treinamentos dos usuários	Relatório de resultados do treinamento dos usuários que foram qualificados para o uso do sistema	Scrum Team
4.6	Reunião de revisão (finalização)	Documentar considerações levantadas na reunião sobre o desenvolvimento das sprints e se o produto final está conforme o TAP, se atende o escopo e se o cliente final necessita de mais alguma alteração	Stakeholders
5.1	Reunião de iniciação	Reunião de início da fase de manutenção e melhorias no sistema, definindo como o cronograma de rotinas será feito e especificação de relatórios de melhorias e bugs encontrados	Stakeholders
5.2	Definição de Rotinas de manutenção e testes	Definição do cronograma e finalidades de manutenção e testes do sistema	Scrum Master
5.3	Definição dos bugs do sistema	Relatório geral com o status de cada bug (se foi ou não resolvido, como foi resolvido, descrição do bug)	Scrum Team
5.4	Definição de resolução dos bugs	Relatório geral das resoluções dos bugs encontrados no sistema	Scrum Team
5.5	Controle de pedidos de alterações	Status de cada pedido, se a melhoria será ou não desenvolvida	Scrum Master

Tabela 2: Enumeração e descrição das atividades vinculados com os papéis (continuação)

	ATIVIDADE	DESCRIÇÃO DE ATIVIDADES	PAPÉIS
5.6	Rastreamento do progresso	Relatório geral com o status de cada melhoria desenvolvida e seus respectivos feedback	Scrum Team
5.7	Medir os resultados das atividades	observações: serve para verificar se as atividades alcançaram realmente o objetivo delas	Scrum Team
5.8	Controle das versões do sistema	Desenvolvimento do documento das versões com os Relatórios de melhorias e resoluções de bugs no sistema, obtendo um controle das versões do sistema	Scrum Master
5.9	Reunião de revisão (transição)	Documentar considerações levantadas na reunião sobre o desenvolvimento das sprints	Stakeholders

3 Execução do projeto

Nesta seção descrevemos o cronograma previsto para o projeto e o estado atual do projeto.

3.1 Backlog e sprints

O cronograma da Tabela 3 abrange todas as sprints que serão realizadas durante o desenvolvimento e as respectivas atividades que deverão ser executadas naquele período. Cada atividade tem como responsável um integrante do projeto. A duração total prevista da execução do processo é de 17 semanas.

Tabela 3: Cronograma com as atribuições aos integrantes

Sprint	Atividade	Integrante	Duração
Estudo de viabilidade	Definição da viabilidade do projeto	Isabelle	1 semana
	Definição do escopo geral	Felipe	
Elicitação e Análise de requisitos	Reunião com o Product Owner	Isabelle	2 semanas
	Definição dos atores	Felipe	
	Análise de requisitos	Isabelle	
	Definição da arquitetura	Nathiely	
Especificação de requisitos	Especificação dos requisitos	Isabelle	1 semana
Validação dos requisitos	Prototipação de interface	Raniel	1 semana
	Avaliar e validar o documento de requisitos	Nathiely	

Tabela 3: *Cronograma com as atribuições aos integrantes (continuação)*

Sprint	Atividade	Integrante	Duração
Estimativas de Prazo	Cálculo de pontos por função	Lucas Taru- moto	1 semana
Projeto Arquitetural	Definição da arquitetura de hardware	Lucas Freitas	1 semana
	Definição da arquitetura de software	Nathiely	
Projeto Detalhado	Definição do modelo de banco de dados	Lucas Taru- moto	2 semanas
	Definição do diagrama de classes	Felipe	
	Definição dos diagramas de atividades	Lucas Freitas	
Implementação	Modelagem da interface com o usuário	Lucas Taru- moto	4 semanas
	Descrição do código	Nathiely	
	Desenvolvimento e codificação	Raniel	
Depuração e Testes	Ciclo de testes	Felipe	2 semanas
	Teste de Componentes	Lucas Freitas	
	Teste de Sistema	Nathiely	
	Teste de Aceitação	Raniel	
Implantação	Manuais do sistema	Lucas Taru- moto	2 semanas
	Instalação	Lucas Freitas	
	Migração	Nathiely	
	Treinamento	Felipe	
Gerência de alteração	Definição de Rotinas de manutenção e testes	Raniel	2 semanas
	Definição dos bugs do sistema	Felipe	
	Definição de resolução dos bugs	Isabelle	
Gerência de pedidos de alteração	Controle de pedidos de alterações	Lucas Freitas	Acompanha o projeto após o sistema ser liberado
Status e medição	Rastreamento do progresso	Lucas Taru- moto	Acompanha o desenvolvimento
Medição dos resultados	Medir os resultados das atividades	Lucas Freitas	Acompanha o desenvolvimento

Tabela 3: *Cronograma com as atribuições aos integrantes (continuação)*

Sprint	Atividade	Integrante	Duração
Gerência de versões	Controle das versões do sistema	Lucas Taru- moto	Acompanha desenvolvimento e a evolução do sistema

3.2 Estado atual

Para o controle dos artefatos gerados pelo processo e das atividades de cada fase foram utilizados os templates abaixo, que têm por objetivo facilitar a visualização do estado atual do projeto, por meio de checklists.

O template da Tabela 4 é o checklist de todos os artefatos divididos por fases.

O template da Tabela 5 é o checklist de todas as atividades divididas por fases.

Ambas as tabelas foram ordenadas por ordem cronológica, facilitando a visualização de quais artefatos ou atividades já foram realizadas e quais ainda precisam ser entregues.

Tabela 4: *Tabela de Checklist de Artefatos*

Levantamento de Requisitos
() Ata de Reunião
() Relatório da viabilidade do projeto
() Termo de Abertura do Projeto
() Relatório de Requisitos iniciais
() Modelo de casos de uso
() Atores
() Descrição da arquitetura
() Relatórios: requisitos funcionais, não funcionais e de negócio
() Documento de Requisitos
() Protótipo Inicial de interface

Tabela 4: *Tabela de Checklist de Artefatos (continuação)*

() Cronograma inicial
() Product Backlog, Ata da reunião
() Ata de Reunião, Sprint Backlog
Projeto e Implementação
() Diagrama de componentes
() Diagrama de pacotes
() Modelo conceitual, Modelo lógico, Dicionário de dados
() Diagrama de classes
() Diagramas de atividades
() Diagrama de implantação
() Protótipo das telas
() Código fonte
() Documento de convenções e padrões
() Ata da reunião, software funcional
() Sprint Backlog, Ata de Reunião
Verificação e Validação
() Caso de teste, Procedimento de teste, Componente de teste, Modelo de teste, Plano de teste, Relatório de testes, Defeito e Avaliação de teste (Teste de Componentes)

Tabela 4: *Tabela de Checklist de Artefatos (continuação)*

() Caso de teste, Procedimento de teste, Componente de teste, Modelo de teste, Plano de teste, Relatório de testes, Defeito e Avaliação de teste (Teste de Sistema)
() Caso de teste, Procedimento de teste, Componente de teste, Modelo de teste, Plano de teste, Relatório de testes, Defeito e Avaliação de teste (Teste de Aceitação)
() Ata da reunião, Software Final e testado
() Sprint Backlog, Ata da reunião
Implantação
() Manual do usuário, Manual do sistema, Termos de uso
() Sistema instalado
() Banco de dados atualizado
() Relatório de usuários qualificados no sistema
() Ata da reunião
() Sprint Backlog, Ata da reunião
Manutenção e Evolução
() Documento do processo de manutenção e testes
() Relatório dos bugs do sistema
() Relatório da resolução dos bugs
() Relatório dos pedidos de alterações
() Relatório das melhorias desenvolvidas
() Relatório do resultados da atividades

Tabela 4: *Tabela de Checklist de Artefatos (continuação)*

<input type="checkbox"/> Documento das versões do sistema juntamente com todas as melhorias e bugs resolvidos
<input type="checkbox"/> Ata da reunião

Tabela 5: *Tabela de Checklist de Atividades*

<input type="checkbox"/> Definição da viabilidade do projeto
<input type="checkbox"/> Definição de escopo geral
<input type="checkbox"/> Reunião com o Product Owner
<input type="checkbox"/> Análise de requisitos
<input type="checkbox"/> Definição dos atores
<input type="checkbox"/> Definição da arquitetura
<input type="checkbox"/> Especificação dos requisitos
<input type="checkbox"/> Avaliar e validar o documento de requisitos
<input type="checkbox"/> Prototipação de interface
<input type="checkbox"/> Cálculo de pontos por função
Projeto e Implementação
<input type="checkbox"/> Definição da arquitetura de hardware
<input type="checkbox"/> Definição da arquitetura de software
<input type="checkbox"/> Definição do modelo de banco de dados
<input type="checkbox"/> Definição do diagrama de classes

Tabela 5: *Tabela de Checklist de Atividades (continuação)*

() Definição dos diagramas de atividades
() Definição do diagrama de implantação
() Modelagem da interface com o usuário
() Desenvolvimento e codificação
() Descrição do código
Verificação e Validação
() Ciclo de testes
() Teste de Componentes
() Teste de Sistema
() Teste de Aceitação
Implantação
() Implementação dos Manuais
() Instalação do sistema
() Migração dos dados
() Treinamentos dos usuários
() Definição de Rotinas de manutenção e testes
() Definição dos bugs do sistema
() Definição de resolução dos bugs
Manutenção e Evolução
() Controle de pedidos de alterações

Tabela 5: *Tabela de Checklist de Atividades (continuação)*

() Rastreamento do progresso
() Medir os resultados das atividades
() Controle das versões do sistema

Complementando, nos anexos possui os templates de relatórios Geral na figura 7, relatório de instalação na figura 8, template de finalização na figura 9, template de Inicialização de Sprint na figura 10 e Template de ata de reunião de definição de Sprints na figura 11

4 Referências bibliográficas

- [1] J. Sutherland, *SCRUM: A arte de fazer o dobro de trabalho na metade do tempo*. Leya, 2014.
- [2] I. Sommerville, “Engenharia de software-8ª edicao (2007),” *Ed Person Education*.
- [3] K. Scott, *O processo unificado explicado*. Bookman, 2003.

5 Anexos

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and extend across the width of the page. There is no handwriting or other markings on the paper.

Observações: _____

_____, de _____ de _____

Figura 7: *Template Relatório Geral*

Número de computadores : _____

[illegible]

_____, de _____ de _____

Template para entrega de texto • Modelo de projeto

Ata de Reunião - Finalização
Data:
Requisitos Alcançados/Conformidade sistema X requisitos:
Considerações sobre segurança/integridade do sistema:
Considerações sobre usabilidade do sistema:
Outras considerações:
<div>Assinatura do Scrum Master</div>

Figura 9: *Template ata de reunião de finalização de fase*

Ata de Reunião - Inicialização do Projeto	
Data:	
Objetivo Geral do Sistema:	
Organização da Equipe:	
Definição das Sprints e tempo de duração:	
Outras Considerações:	
<hr/>	
Assinatura do Scrum Master	

Figura 10: *Template ata de reunião de inicialização do Projeto*

Ata de Reunião - Sprint	
Data:	
Fase/Sprint:	
Atividades Concluídas:	
Atividades Pendentes/Incompletas:	
Problemas/Desafios Encontrados:	
Considerações para a próxima Sprint:	
<hr/>	
Assinatura do Scrum Master	

Figura 11: *Template ata de reunião de iniciação de Sprint*