

# Universidade Federal de Ouro Preto



## Engenharia de Software II

Sistema de *Urna Eletrônica*

Grupo: *Unsigned*

Alunos: Bernardo Alexandre Santos Emery, Danielle Emely de Souza Almeida,  
Eduardo Lopes Amaro, João Pedro Siqueira Mendes,  
Lamarck Ferreira da Silva, Nicolas Vasca Galindo, Wesley Claudio Dias.  
Professor: Msc prof. Johnatan Oliveira  
Horário: Seg & Qua - 08:20 -10:00

Ouro Preto, 06 de Janeiro de 2021

# Conteúdo

<b>1</b>	<b>Histórico de Revisões</b>	<b>1</b>
<b>2</b>	<b>Processo e Software</b>	<b>1</b>
<b>3</b>	<b>Cronograma</b>	<b>1</b>
<b>4</b>	<b>Levantamento de Requisitos</b>	<b>2</b>
<b>5</b>	<b>Especificação de Requisitos</b>	<b>3</b>
5.1	Requisitos Funcionais . . . . .	3
5.2	Requisitos Não Funcionais . . . . .	3
<b>6</b>	<b>Plano de VVT</b>	<b>4</b>
6.1	Requisitos a serem testados . . . . .	4
6.2	Estratégias e ferramentas de teste . . . . .	5
6.3	Equipe e infra-estrutura . . . . .	5
6.4	Execução do Plano de Teste . . . . .	6
<b>7</b>	<b>Medição e Qualidade de Software</b>	<b>7</b>
<b>8</b>	<b>Observações</b>	<b>7</b>
<b>9</b>	<b>Referências</b>	<b>8</b>

## 1 Histórico de Revisões

Data	Versão	Descrição	Autor
10/03/2021	1.0	Criação Deste Documento	João Pedro e Lamarck da Silva
10/03/2021	1.1	Inclusão de Requisitos e de Entidades	João Pedro e Lamarck da Silva
10/03/2021	2.0	Planning de Validação	João Pedro e Lamarck da Silva
01/04/2021	2.1	Criação do Plano VVT	Eduardo Lopes Amaro

Tabela 1: Revisões do Documento

## 2 Processo e Software

Foi escolhido como modelo de processo de software o método : **Agile kabam**. O Metodo Agile kabam, trabalha em colaboração com o cliente, estabelecendo metas factíveis e apurando paulatinamente o progresso da transformação através evidências que demonstrem os resultados alcançados.

## 3 Cronograma

Nome	Tarefa	Prazo
Lamarck Ferreira da silva	PRODUCT OWNER	10/03 a 25/04
João Pedro Mendes	PRODUCT OWNER	10/03 a 25/04
Nicolas Vasca Galindo	BACK-END	10/03 a 25/04
Wesley Dias	BACK-END	10/03 a 25/04
Eduardo Lopes	TESTER	10/03 a 25/04
Bernardo Alexandre	FRONT-END	10/03 a 25/04
Danielle Almeida	FRONT-END	10/03 a 25/04

Tabela 2: Cronograma

## 4 Levantamento de Requisitos

Para o levantamento de requisitos foram analisadas as eleições municipais, estaduais e presidenciais. Além disso, foram analisadas as eleições de reitores, representantes de colegiados e representantes de turma. Após consolidar os dados obtidos foram definidas as entidades do sistema. Ao consolidar os mecanismos utilizados para computar os votos, pudemos definir os requisitos funcionais do sistema. Após reunir com o grupo envolvido foram definidos os requisitos não funcionais

Para o levantamento de requisitos, o analista dispõe de algumas técnicas que são utilizadas de acordo com o perfil do cliente. Existem diversas técnicas, cada uma adequada para um cenário específico, e dentre as comumente utilizadas podemos citar as seguintes técnicas:

### 1. Descoberta de Requisitos (Pontos de vista) :

Na democracia representativa, a eleição é o processo que consiste na escolha de determinados indivíduos para exercerem o poder soberano, concedido pelo povo através do voto. Com isso se dá a necessidade de um mecanismo para computar os votos. A urna eletrônica vem como solução para a necessidade de computar esses votos.

### 2. Casos de Uso :

Júlio deseja votar no candidato "João", porém Júlio não sabe o código de João. Ao chegar para votar, Júlio informou os seus dados ao mesário e seguiu para a urna. Ao informar seus dados, Júlio, selecionou listar todos os candidatos, com isso obteve o código de João. Júlio selecionou votar, digitou o código de João e confirmou.

Bernardo é um cidadão que se voluntariou como mesário para as próximas eleições. Ao receber suas funções uma delas era popular a urna com os candidatos daquela eleição e os eleitores, da sua respectiva zona eleitoral.

José se candidatou para a próxima eleição, ao se candidatar, José informou seu nome completo, e recebeu um código, para fazer sua campanha eleitoral. Agora José aguarda o resultado da eleição.

Daniele, deseja votar em José, e obteve o seu número durante a campanha eleitoral. Ao chegar para votar, Daniele informou os seus dados ao mesário e seguiu para a urna. Ao informar seus dados, Daniele, selecionou votar agora. Digitou o número de José e confirmou.

## 5 Especificação de Requisitos

### 5.1 Requisitos Funcionais

Requisitos Funcionais são assuntos de importância fundamental ou essencial ao produto. Eles descrevem o que o produto tem de fazer ou que ações processuais deve tomar.

**RF01–Cadastros** O mesário deve conseguir cadastrar os candidatos e os eleitores daquela eleição.

**RF02– Votar** O eleitor deve conseguir entrar com os seus dados, listar os candidatos disponíveis para aquela votação e votar em um candidato.

**RF03– Computar Votos** Ao final da eleição o mesário deverá conseguir listar os candidatos e quantos votos cada um recebeu, a listagem deverá ser por ordem de votos.

**RF03– Tela de Votação** A votação será realizada via interface gráfica que simula os elementos de uma urna.

### 5.2 Requisitos Não Funcionais

Requisitos Não Funcionais são as propriedades que as funções devem ter, tais como desempenho e usabilidade. Não se detenha ao seu nome pouco apropriado (nós o usamos porque é a maneira mais comum de se referir a estes tipos de requisitos)—estes requisitos são tão importantes quanto as exigências funcionais, para o sucesso do produto.

**RNF01.Linguagem de desenvolvimento do Front-End:** TypeScript

**RNF02.Linguagem de desenvolvimento do Back-End:** JavaScript

**RNF03. Framework:** REACT + EXPRESS

**RNF04. Banco de Dados:** MongoDB

## 6 Plano de VVT

O objetivo do plano de VVT é assegurar que o software cumpra com suas especificações e atenda às necessidades dos usuários. A ideia é testar as funcionalidade e o desempenho da arquitetura do sistema selecionada garantindo que ocorram corretamente.

### 6.1 Requisitos a serem testados

Os itens a seguir identificam o conjunto de requisitos a serem testados no projeto, apresentando o caso de uso, requisitos funcionais e não funcionais que foram identificados como alvos do teste.

Nos testes de interface com o usuário vamos verificar os seguintes subsistemas:

1. Cadastrar um eleitor: "O usuário com os dados do eleitor deve ser capaz de cadastrá-lo no sistema".
2. Cadastrar um candidato: "O usuário com os dados do candidato deve ser capaz de cadastrá-lo no sistema".
3. Visualizar todos os candidatos: "O eleitor deve ser capaz de visualizar todos os candidatos disponíveis e seus respectivos códigos".
4. Votar em um candidato: "O eleitor deve ser capaz de inserir o código do candidato e confirmar seu voto".

Nos testes de desempenho vamos verificar as seguintes medidas:

1. Tempo de resposta para um eleitor realizar login: "Verificar quanto tempo foi gasto para um eleitor realizar login no sistema".
2. Tempo de resposta para visualizar todos os candidatos: "Verificar quanto tempo foi gasto para um eleitor visualizar todos os candidatos disponíveis no sistema".
3. Tempo de resposta quando ocorrer 1000 acessos ao mesmo tempo: "Verificar se o sistema suporta um grande número de acessos simultâneos".

Nos testes de segurança e controle de acesso vamos verificar os seguintes subsistemas:

1. Verificar o login do usuário: "Após cadastrado, verificar se um eleitor pode fazer login no sistema".

2. Verificar se o CPF informado é válido: "Ao digitar o CPF para realizar login, verificar se o CPF digitado é válido".

## 6.2 Estratégias e ferramentas de teste

Para realização dos testes as seguintes estratégias foram adotadas.

Nos testes de interface com o usuário vamos verificar a interação do eleitor com o sistema, com o objetivo de assegurar que a interface forneça o acesso e a navegação adequados por meio das funções do sistema.

**Objetivo do teste:** Verificar a navegação do eleitor pelo sistema e se os métodos desenvolvidos cumprem com o esperado.

**Técnica:** Criar e modificar testes para cada janela a fim de verificar a navegação e os estados de cada interação.

**Critério de Conclusão:** Verificar com êxito se todas as tarefas realizadas foram concluídas.

Para os testes de desempenho vamos verificar o tempo de resposta do sistema para a realização de uma tarefa do eleitor e o tempo de resposta do sistema com um pico de acessos, com o objetivo de verificar se os requisitos de desempenho foram alcançados.

**Objetivo do teste:** Validar o tempo de resposta do sistema para as tarefas realizadas pelo eleitor.

**Técnica:** Utilizar os scripts de teste desenvolvido para testar o tempo de resposta do sistema.

**Critério de Conclusão:** Concluir com êxito os scripts de teste para um ou vários usuários sem nenhum erro e com tempo aceitável.

Nos testes de segurança e controle de acesso vamos verificar a segurança do sistema, com o objetivo de assegurar que apenas os usuários cadastrados sejam capazes de acessar e realizar alguma ação.

**Objetivo do teste:** Verificar se o eleitor pode acessar apenas as funções e dados nos quais ele possui permissão e se os dados informados estão corretos.

**Técnica:** Identificar todos os tipos de usuários e verificar seu nível de permissão.

**Critério de Conclusão:** Para cada usuário identificado, se os dados forem corretos e a função estiver disponível para seu nível de permissão, a tarefa pode ser finalizada corretamente.

## 6.3 Equipe e infra-estrutura

Contém descrição da equipe e da infra-estrutura utilizada para o desenvolvimento das atividades de testes, incluindo: pessoal, equipamentos, software

de apoio, materiais, dentre outros. Isto visa garantir uma estrutura adequada para a execução das atividades de testes previstas no plano.

## 6.4 Execução do Plano de Teste

<b>Funcionalidade:</b> Envelope		<b>Tempo Despendido(h):</b> 1h
<b>Contador:</b> 02		<b>Criticidade:</b> Baixa

  

<b>Objeto de Teste:</b>		
<ul style="list-style-type: none"> <li>➤ Validar uso do assistente para criar documento no modelo Envelope.</li> <li>➤ Verificar impressão usando um envelope.</li> </ul>		
<b>Descrição do Caso de Teste:</b>		
<ul style="list-style-type: none"> <li>➤ O aplicativo deverá atender corretamente a configuração fornecida pelo usuário (nas 3 abas: Envelope, Formato e Impressora).</li> <li>➤ A impressão no envelope deverá ser realizada com sucesso.</li> </ul>		
<b>Pré-Condição:</b>		
<ul style="list-style-type: none"> <li>➤ O usuário deverá informar dados do Destinatário e Remetente.</li> <li>➤ Existir uma impressora compatível para impressão no envelope.</li> </ul>		
<b>Dados de Entrada:</b>		
ID	Passo	Procedimento
1	P1	Executar o aplicativo LibreOffice 4.2 (Opção texto)
	P2	Selecionar a opção Texto
	P3	Acessar o menu a opção Envelope (Inserir>>Envelope)
	V1	O aplicativo deverá exibir a tela do assistente para Envelope
	P3	Preencher os dados do campo Destinatário e Remetente
	P4	Clicar no botão Inserir
2	P5	Se necessário ajustar a largura da caixa de texto do Destinatário e Remetente
	P7	Clicar no ícone da Impressora
	V2	Verificar se o envelope foi impresso corretamente.
<b>Resultado Esperado:</b> As operações deverão funcionar corretamente cumprindo todas as regras acima citadas.		

Figura 1: Exemplo



## 7 Medição e Qualidade de Software

Apresente aqui o formato da Medição e qualidade de software. Você deve mostrar os meios que irá avaliar a qualidade do seu software. Apresente o plano e os resultados a partir da prática de ferramentas de detecção de code smells, por exemplo. Em Java, temos uma ferramenta chamada JDEODORANT. Você pode avaliar as métricas de qualidade também, por exemplo, em Java, temos CKMetrics<sup>1</sup>

## 8 Observações

Apresente aqui as dificuldades na disciplina, trabalho prático e coisas do tipo.

---

<sup>1</sup><https://github.com/mauricioaniche/ck>

## 9 Referências

- [1] Chapman, S.J. – Electric Machinery Fundamentals, 4th Edition;
- [2] Fitzgerald, A. E. – Máquinas Elétricas, 2da Edição;