#### Relatório

#### Fundamentos da Engenharia de Software

Integrantes: Joyce Brum,

Thiago Outeiro,

Gabriel Mattos e

Vitor Millioni

**Sumário**

|  |  |
| --- | --- |
| 1. Time |  |
| 1. Objetivo |  |
| 1. Planejamento Inicial: I-Educar |  |
| 1. Mudanças no planejamento: Merenda |  |
| * 1. O que? |  |
| * 1. Por que? |  |
| * 1. Quem? |  |
| * 1. Como? |  |
| * 1. Quando e quanto? |  |
| 1. Desenvolvimento |  |
| 1. Ferramentas |  |
| 1. Conclusão |  |
| Anexo I – Histórias de usuário |  |
| Anexo II – Metáfora do Barco |  |
| Anexo III – Casos de Uso |  |
| Anexo IV – Casos de Teste |  |
| Anexo V – Testes Documentados |  |
|  |  |

**Time**

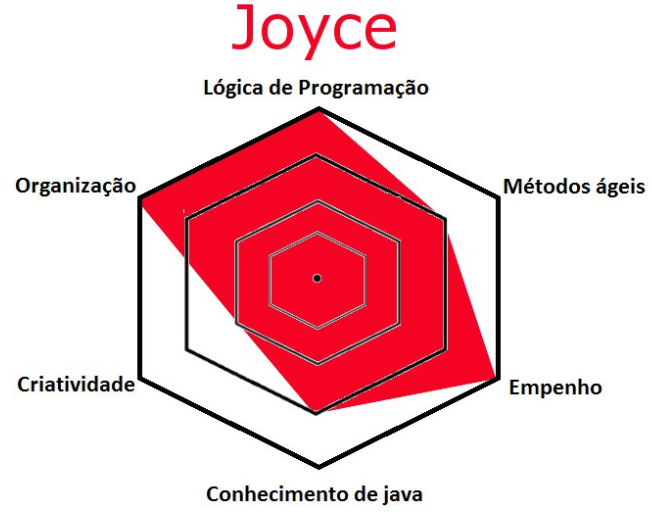
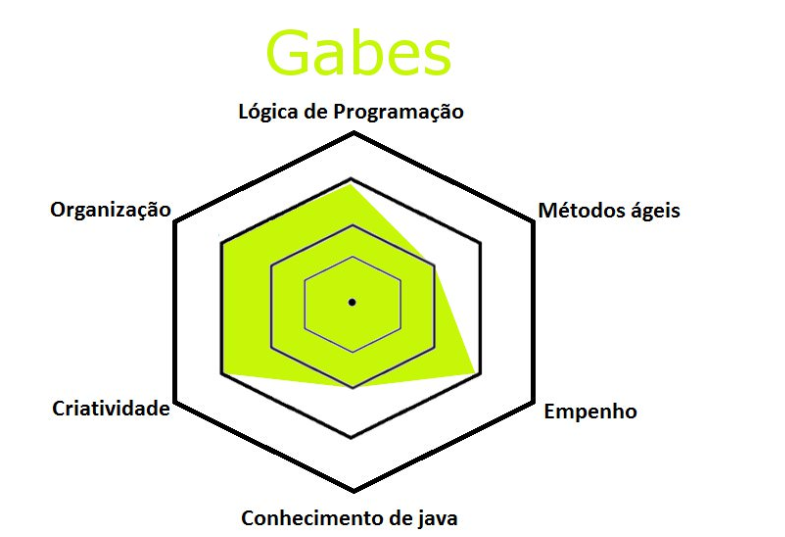
Decidimos dar um trabalho com algum enfoque para cada membro do time. Thiago, Gabriel e Joyce ficaram como desenvolvedores enquanto que Vitor ficou responsável pelo Banco de Dados e pelos testes.

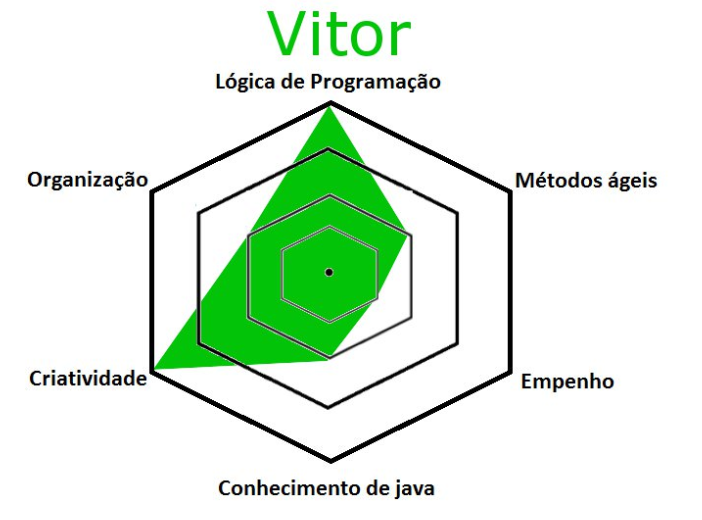
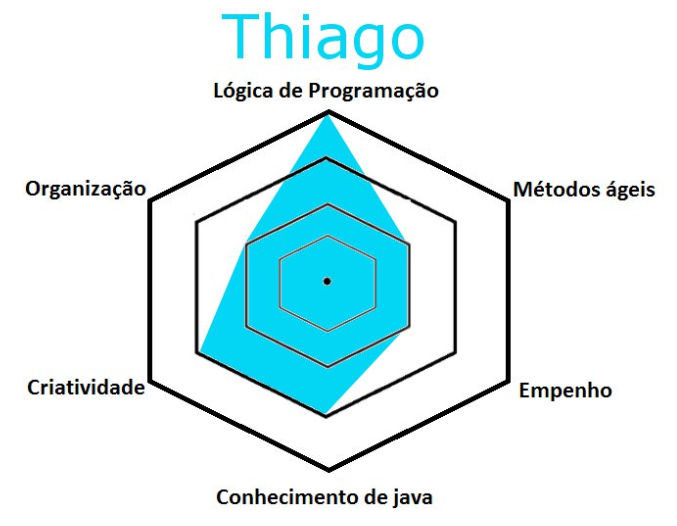
Thiago ficou com um foco maior na interface do projeto, desenvolvendo as telas e planejando o modelo de interface. Gabriel ficou responsável por atualizar a wiki, além de ajudar na programação. A Joyce atuou também como Scrum Master, sendo responsável por coordenar as sprints e acompanhá-las.

**Conhecimentos prévios:**

Todos os membros do time já possuíam algum conhecimento de java devido à matéria de computação 2, mas algumas bibliotecas necessárias para a aplicação exigiram uma atenção maior de aprendizado.

Sobre os conhecimentos principais para o desenvolvimento do software e aprendizados na disciplina escolhemos os 6 principais: Lógica de programação, Criatividade, Organização, Métodos ágeis, Empenho, Conhecimento de Java.



****

**Objetivo**

O projeto inicial que nos foi proposto para trabalhar foi o de customização de relatórios para uma plataforma online chamada i-Educar. A fim de nos deixar mais próximos de entender e conseguir corretamente estimar nossos objetivos, requisitos e motivações, fizemos um conjunto de reuniões com um representante do i-Educar e com a prefeitura de Caxias.

Diante dos fatos apresentados, surgiu então uma nova ideia de projeto: Merenda. Tal projeto seria focado a auxiliar a prefeitura de caxias no controle das merendas através de um sistema de geração de relatórios segundo o modelo mostrado.

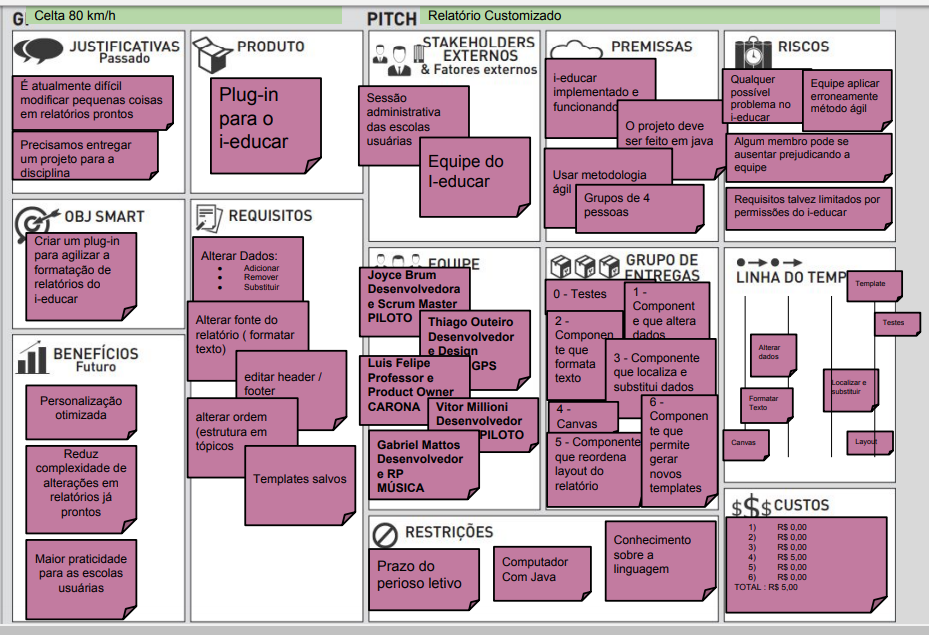
Nosso time escolheu então pela opção da Merenda, mas como boa parte do tempo já havia passado, optamos por não refazer oficialmente o planejamento todo que havia sido feito para a customização de relatórios.

Ao conversar com a representando da prefeitura de caxias sobre os critérios que o software deveria atender, concluiu-se então, que o principal objetivo seria a geração de relatório mensal por escola, com possibilidades de atualização diária dos dados, com os dados da escola, o cardápio do mês e a lista de itens usados no mesmo.

**Planejamento Inicial: I-Educar**

A princípio, o projeto era sobre customizar relatórios para o i-Educar. Para isto, fizemos um canvas a fim de planejar o base do projeto. Neste canvas, respondemos às 6 perguntas fundamentais para o desenvolvimento e planejamento de um projeto: “Por quê?”, “O que?”, “Quem?”, “Como?”, “Quando?” e “Quanto?”, separado em colunas e poucos subtópicos a fim de tornar o processo de idealização e planejamento menos maçante e mais iterativo.

Tendo em vista este objetivo, geramos o canvas abaixo:

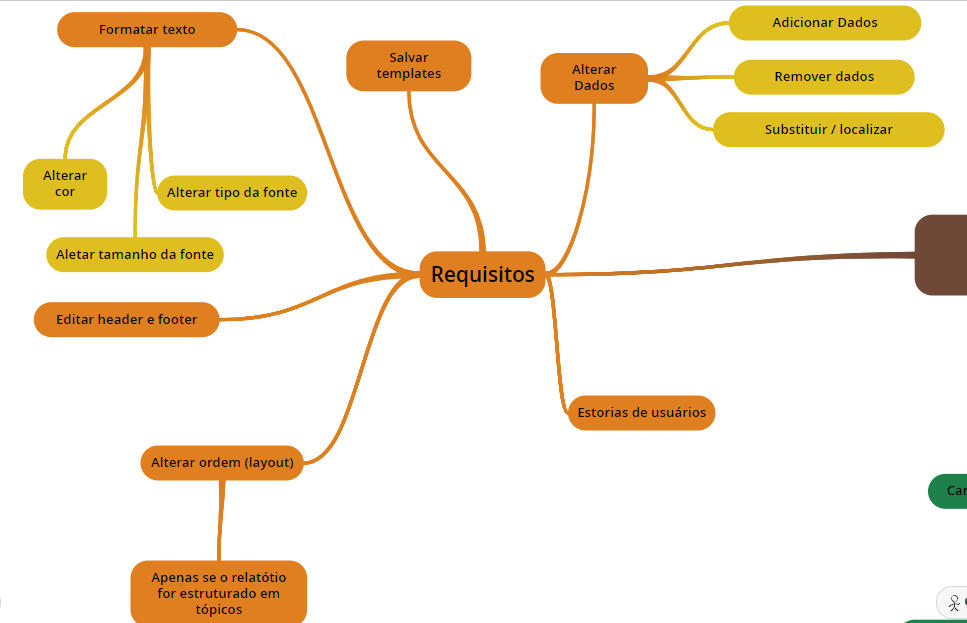


Além do canvas, que mostrava de uma maneira geral e organizada os principais tópicos do planejamento do software, fizemos um mapa mental com o objetivo de mapear de uma maneira mais visual os requisitos, limitações e objetivos ilustrados no canvas acima.

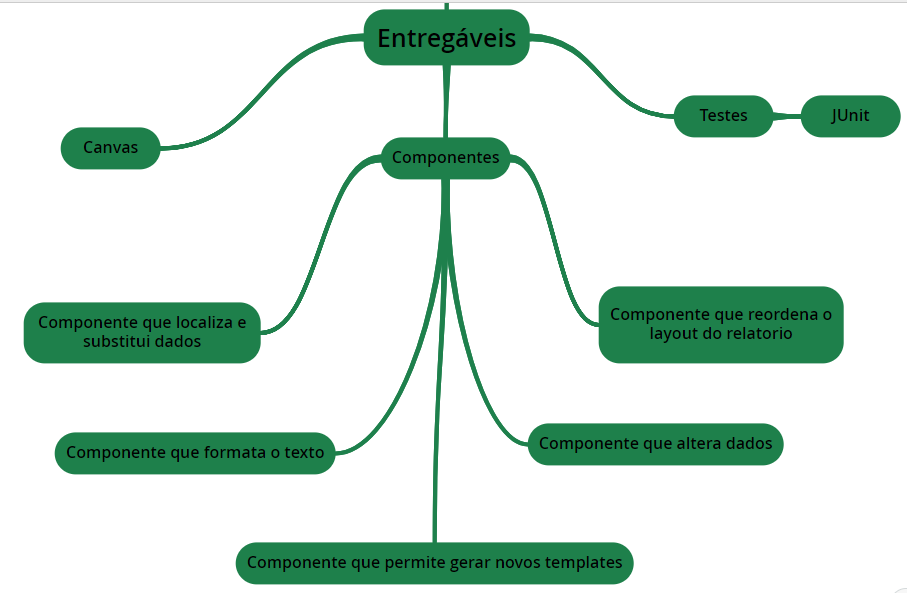


O mesmo possui quatro subárvores principais:

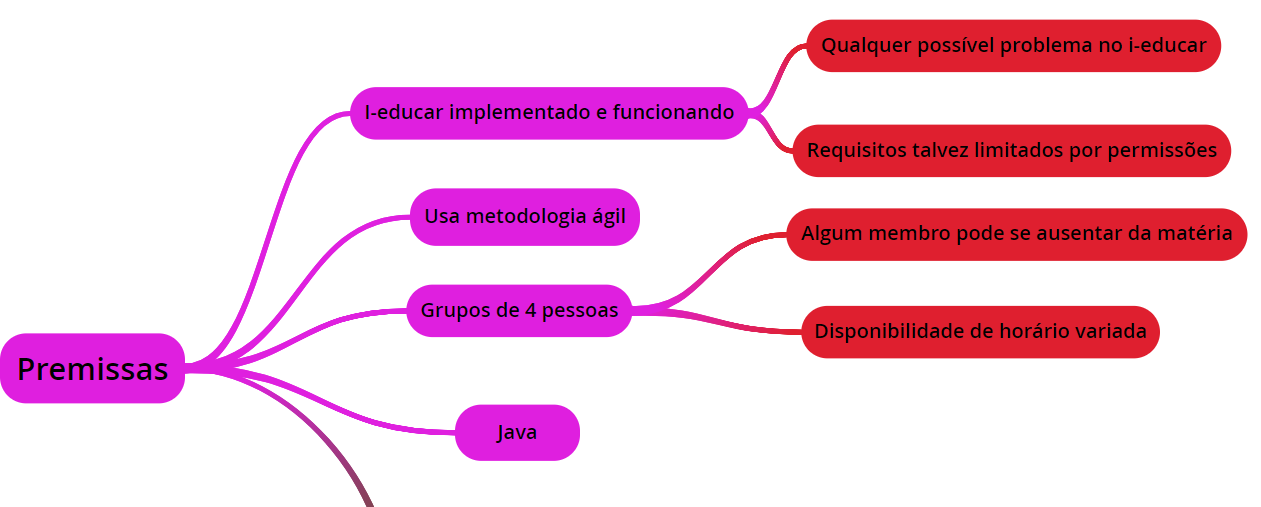
**Requisitos**:



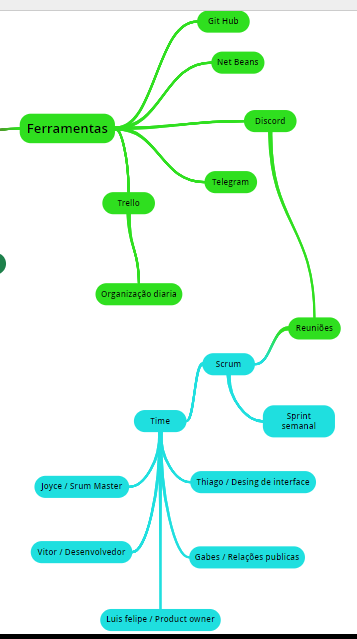
**Entregáveis**



**Premissas**

****

**Ferramentas**

****

**Mudanças no Planejamento:**

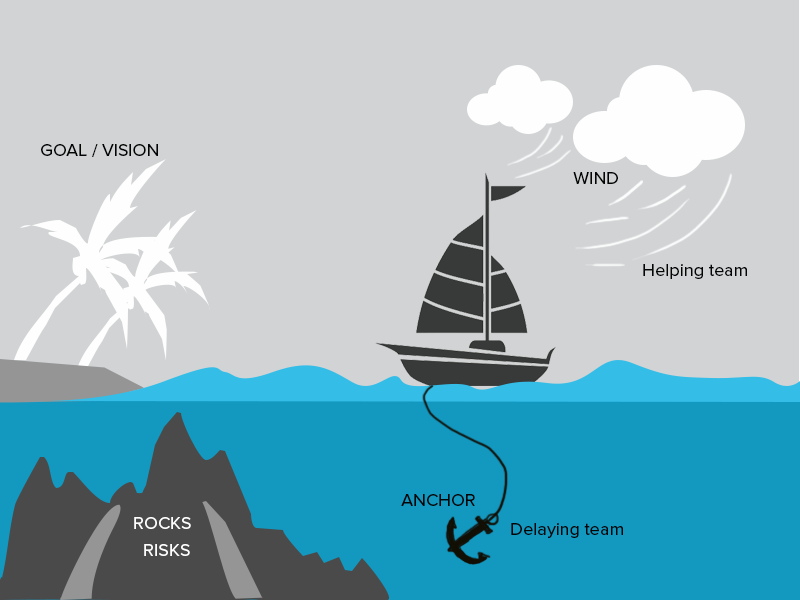
**Merenda**

A parte de ferramentas continuou a mesma, entretanto, as demais mudaram bastante. As premissas se mantivera, apenas a que exigia que o i-Educar estivesse implementado e funcionando não era mais necessária, visto que o novo projeto não possuía nenhum tipo de dependência com esta plataforma.

Os entregáveis e requisitos passaram a ser baseados nas *histórias de usuário* desenvolvidas em conjunto com a prefeitura em caxias e que podem ser vistas com mais detalhes no Anexo I. As histórias tiveram um importante papel no desenvolvimento do software, pois elas representavam de maneira simples e clara qual a funcionalidade desejada do usuário para a aplicação. E como a linguagem era simples, a validação da mesma com o *product owner* era muito mais transparente.

Elas foram a base para as tarefas que escolhíamos e designávamos a cada sprint.

A fim de poder identificar os principais fatores influenciadores em nosso projeto, utilizamos a metáfora do barco associando a cada um dos componentes os aspectos que achávamos que melhor se encaixavam no nosso contexto.

Todas as associações e explicações, sobre a metáfora, do barco podem ser vistas no Anexo II.

***O que?***

Um software funcional que auxilie no controle de merendas nos colégios da rede pública, seguindo como exemplo a prefeitura de caxias. Tal software deve atender aos requisitos necessários para a geração de relatórios de controle da merenda escolar.

***Por que?***

Todo o controle de merenda da rede pública é feito via excel, o que torna o lançamento de dados de merenda e até a verificação de tais dados trabalhosos. O software visa facilitar a geração, visualização e controle dos relatórios gerados por escola, de forma a automatizar parte do trabalho e centralizar a informação em um banco de dados.

**Quem?**

O software vai ser desenvolvido pelos membros do time citados e apresentados no tópico “Time”, além da participação do product owner Luis Felipe Coimbra Costa. Contamos também com a presença e participação da Poliana Guedes, representante intermediária da prefeitura de caxias, auxiliando-nos na criação e validação dos requisitos funcionais do software afim de atender da melhor maneira possível à demanda.

Como?

Utilizamos como especificação de requisitos as histórias de usuário, que contam de maneira resumida e informal uma breve história da aplicação de alguma funcionalidade do software informando o tipo de usuário que utilizaria aquela funcionalidade, a funcionalidade em si e sua motivação. Todas as histórias de usuário produzidas pela equipe podem ser vistas no Anexo I

***Quando e Quanto?***

A cada semana, escolhemos algumas histórias para trabalhar, a fim de alcançar aquele requisito. Ao final da semana, as tarefas determinadas para a história devem estar completas de forma a que ao final das 11 semanas, todos os requisitos estivessem implementados e testados. O custo do projeto foi de 5 reais, gastos em papeis, post it’s e outros objetos de papelarias usados para planejamento do mesmo.

**Desenvolvimento**

Utilizando a técnica de métodos ágeis, realizamos um total de 11 sprints, em que em cada uma delas avaliávamos as tarefas das histórias que cumprimos na última sprint e apresentávamos nossas dificuldades e desenvolvimento no projeto. Em seguida, selecionávamos um conjunto de histórias para trabalhar naquela sprint e de cada uma delas selecionávamos um conjunto de tarefas. Cada tarefa ganhava uma estimativa de acordo com o seu nível de dificuldade.

Usando as estimativas, tentávamos deixar a distribuição de trabalho bem distribuída entre os membros, levando sempre em conta a disponibilidade de cada um.

Participamos também de uma atividade proposta pelo Luis que se tratava de passar um dia em um ambiente de desenvolvimento de software de uma empresa que trabalha com métodos ágeis. Lá, utilizamos da estratégia Pomodoro para fazer uma sequência de tarefas com tempos predeterminados, a fim de que a sensação de estar cumprido com o programado ser uma constante e nos estimular mais.

  
Illustration 1: Pomodoro

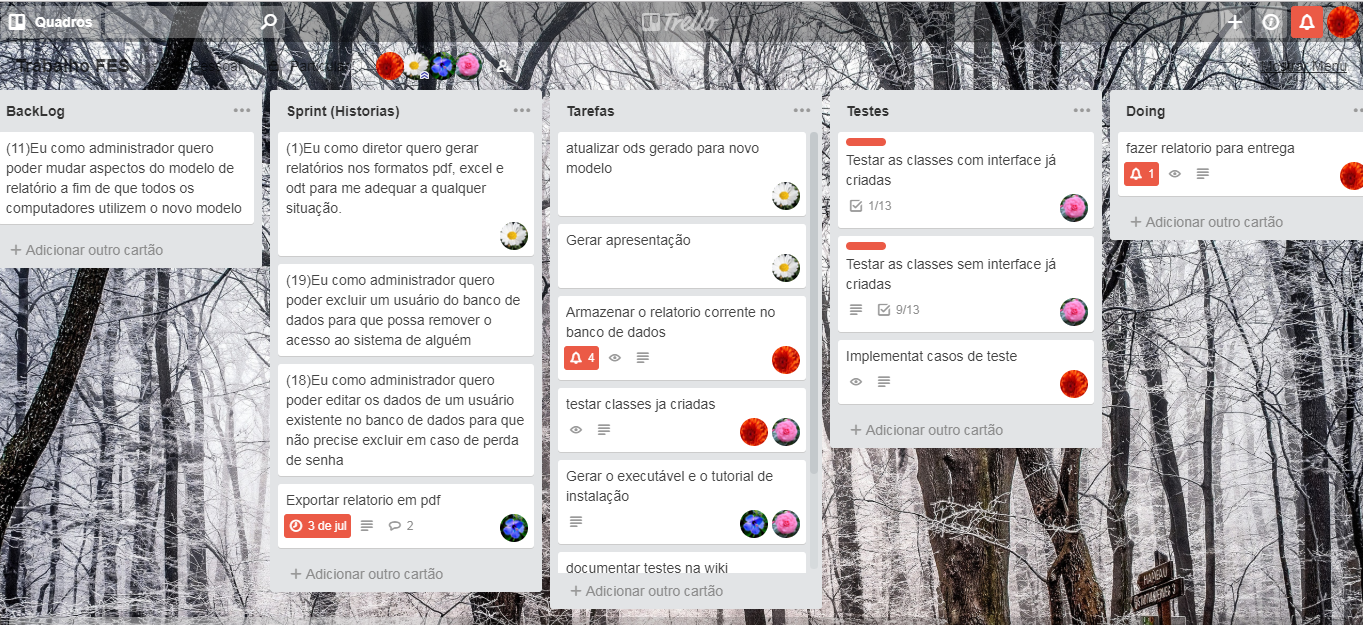
Essa atividade foi muito importante para começarmos de vez com a implementação dos testes, em específico os testes unitários. O ambiente agradável do local também foi um forte estímulo para que nos empenhássemos ainda mais em completar as tarefas.

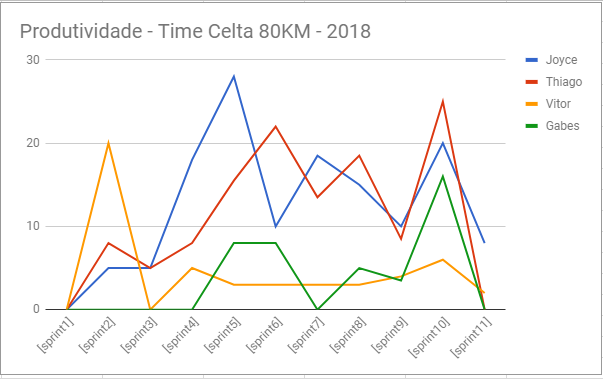
Conseguimos implementar um banco de dados funcional, o que foi muito satisfatório tendo em vista que era uma das dificuldades maiores que previmos para o projeto. A cada sprint, sempre que era possível encontrar a representante de caxias, mostrávamos o que já tínhamos pronto até o momento, pedindo a ela feedback sobre o que poderia melhorar ou revendo algumas histórias para confirmá-las.

Quanto aos testes, os testes realizados e seus resultados estarão anexados no Anexo IV.

**Ferramentas**

Para organização da Sprint e acompanhamento da mesma utilizamos o trello.

  
Illustration 2: Trello para Sprint

As sprints eram feita presencialmente às quartas, mas os *daily meeting* era feito pelo telegram, onde compartilhávamos o nosso progresso e dificuldades. Para acompanhar o nosso desenvolvimento individual no decorrer da sprint (da melhor maneira possível visto que a mesma poderia estar sujeita a erros de estimativa) utilizamos o burndown chart pelo excel online.

**Conclusão**

Com o decorrer do projeto, todos os membros evoluíram seus conhecimentos em relação à distribuição inicial. Sendo colocado na prática, os conhecimentos em métodos ágeis se fizeram entender, todas as dinâmicas, as entregas semanais, as sprints, tudo que antes era apenas um brain storm associado a métodos ágeis, passou a ser parte uma estratégia coesa e coerente na mente de cada um dos membros.

Sobre conhecimento em java, a descoberta de novas API’s e bibliotecas e a dificuldade em aprender a usá-las foram fatos constantes no desenvolvimento do software, aumentando consideravelmente o conhecimento dos programadores sobre java.

Vale salientar também que a aplicação de testes na prática foi o fator determinante para que entendêssemos sua importância no processo de engenharia de software. Através dos testes foi possível identificar tanto erros de programação quanto erros de arquitetura de classes, o que permitiu um aprimoramento intenso do código e da própria aplicação.

A presença quase que constante do usuário nas sprints mostrou a importância da participação do cliente no processo da engenharia de software baseada em métodos ágeis, pois com isso foi possível, a todo instante, verificar se de fato o que estava sendo desenvolvido atendia à demanda do cliente e, caso não atendesse, mudar a tempo a fim de evitar maiores prejuízos ao orçamento e ao prazo de entrega.

**ANEXO I**

**HISTÓRIAS DE USUÁRIO**

1. Eu como diretor quero gerar relatórios nos formatos pdf, excel e odt para me adequar a qualquer situação.
2. Eu como diretor quero poder gerar o relatório de maneira sequencial para ter mais praticidade.
3. Eu como diretor quero poder lançar atualizações diárias para poder não ter todo o trabalho de uma vez.
4. Eu como diretor quero poder editar dados já lançados para que não tenha que gerar um novo relatório do zero em caso de erro.
5. Eu como diretor quero poder me referenciar apenas ao mês e ano para gerar a página do cardápio para que não tenha que olhar no calendário.
6. Eu como diretor quero cadastrar os dados da minha escola de forma a não ter que ficar repetindo-os em todos os relatórios.
7. Eu como diretor gostaria que cálculos usando dados pré-inseridos fossem feitos pelo sistema para que eu tenha mais praticidade.
8. Eu como diretor gostaria de lançar diretamente os itens usados no cardápio para que não gaste tempo procurando um a um na tabela.
9. Eu como administrador quero poder mudar aspectos do modelo de relatório a fim de que todos os computadores utilizem o novo modelo
10. Eu como leitor quero poder filtrar os relatórios por escola para facilitar minha busca
11. Eu como leitor quero poder filtrar os relatórios por mês de referência para facilitar minha busca
12. Eu como usuário do sistema gostaria de poder ter acesso ao sistema através de um login e senha a fim de que possa trabalhar efetivamente.
13. Eu como administrador quero poder gerar novos perfis a fim de que possa dar acesso a outras pessoas de acordo com seu nível de acesso.
14. Eu como usuário gostaria que todos os arquivos fossem salvos em um banco de dados para centralizar a informação.
15. Eu como usuário quero poder interagir com uma tela de login a fim de que a aplicação fique mais intuitiva e organizada
16. Eu como administrador quero adicionar e modificar permissões por tipo de usuário para que tenha mais liberdade de modificar futuramente.
17. Eu como administrador quero poder editar os dados de um usuário existente no banco de dados para que não precise excluir em caso de perda de senha
18. Eu como administrador quero poder excluir um usuário do banco de dados para que possa remover o acesso ao sistema de alguém

**ANEXO II**

**METÁFORA BARCO**

Elementos associados ao celta: Âncora – Freio de mão; Rochas – Buraco na Estrada; Vento – Motor

FREIO DE MÃO

* Nenhum dos membros do time tem conhecimentos de banco de dados.
* Dois dos membros do time estagiam, limitando os horários de reunião.
* Um dos membros tem 6 matérias puxadas.

BURACO NA ESTRADA

* Provas de outras matérias atrapalham o desenvolvimento.
* Não conseguirmos implementar o banco de dados.
* Ficar 200% desmotivados

MOTOR

* Gabriel possui conhecimento em interfaces em java.
* Todos os membros possuem bons conhecimentos de lógica de programação e programação orientada à objetos.
* Estamos 200% motivados.
* Comunicação do time é impecável.

OBJETIVO

* Uma aplicação funcional que possibilite o controle de merendas pelas escolas usuárias, com um banco de dados interno.
* Conseguir um bom desempenho na matéria com nosso desenvolvimento individual.

**ANEXO III**

**CASOS DE USO**

|  |
| --- |
| **CASO DE USO 1** |
| 1. **Nome do caso de uso:** |
| Acesso ao sistema |
| 1. **Descrição resumida:** |
| Caso de uso referente ao login do usuário no sistema |
| 1. **Ator:** |
| Usuário |
| 1. **Atores Secundários:** |
| não há |
| 1. **Pré-condição:** |
| Usuário precisa estar cadastrado no sistema |
| 1. **Fluxo de eventos** |
| * 1. Digita o login de acesso |
| * 1. Digita senha de acesso |
| * 1. Clicka em logar |
| 1. **Pós-condições:** |
| Sistema deve redirecionar para a tela principal |
| 1. **Fluxos Alternativo** |
| * 1. Usuário pode ter errado seu login |
| * + 1. Aparece um pop-up dizendo que o nome de usuário está incorreto. |
| * 1. Usuário errou sua senha |
| * + 1. Aparece um pop-up informando que a senha está incorreta. |
| 1. **Fluxo de Exceção**: |
| Não há |

|  |
| --- |
| **CASO DE USO 2** |
| 1. **Nome do caso de uso:** |
| Ver Relatório |
| 1. **Descrição resumida:** |
| Usuário pode ver um relatório filtrado por mes e ou ano em pdf , de acordo com o contexto de escola em que se encontra. |
| 1. **Ator:** |
| Usuário |
| 1. **Atores Secundários:** |
| não há |
| 1. **Pré-condição:** |
| Estar logado no sistema e ter a permissão para ver relatório |
| 1. **Fluxo de eventos** |
| * 1. Seleciona a opção lista de relatórios |
| * 1. Digita no campo “ano” o ano do relatório desejado |
| * 1. Digita no campo “mês” o mês do relatório desejado |
| * 1. A lista de relatórios irá mostrar apenas o relatório do mês e do ano desejados. |
| * 1. Selecione o relatório. |
| * 1. Selecione a opção ver relatório |
| * 1. Selecione o formato do relatorio (odt, pdf ou ods) |
| * 1. Selecione a opção abrir |
| Sistema deve abrir o documento (pdf, odt ou ods) mostrando o relatório selecionado |
| 1. **Fluxos Alternativo** |
| * 1. Usuário só informou um dos campos (mês ou ano) |
| * + 1. Em vez de aparecer apenas um relatório, aparecerá a lista filtrada pelo campo informado. |
| 1. **Fluxo de Exceção**: |
| * 1. Não existir relatório encontrado para o filtro informado |
| * + 1. A opção ver relatório não será habilitada |
| * 1. Relatório não encontrado no banco de dados |
| * + 1. Informa ao usuário que houve um problema com o banco de dados e que o relatório não está disponível para visualização |

|  |
| --- |
| **CASO DE USO 3** |
| 1. **Nome do caso de uso:** |
| Editar relatório |
| 1. **Descrição resumida:** |
| Usuário poderá editar o conteúdo de relatórios já criados, de qualquer mês ou ano de referência, desde que o mesmo seja da escola a qual ele é responsável. |
| 1. **Ator:** |
| Usuário |
| 1. **Atores Secundários:** |
| não há |
| 1. **Pré-condição:** |
| Estar logado no sistema e ter a permissão para editar relatório |
| 1. **Fluxo de eventos** |
| * 1. Seleciona a opção editar relatório |
| * 1. Escolhe um relatório dentre os mostrados na lista |
| * 1. Seleciona a opção avançar |
| * 1. Edita os dados da capa |
| * 1. Avançar |
| * 1. Edita os dados do cardápio |
| * 1. Avançar |
| * 1. Edita os dados da lista de ingredientes |
| * 1. Seleciona salvar para salvar as modificações no banco de dados. |
| 1. **Pós-condições:** |
| Sistema deve salvar as modificações feitas no relatório selecionado no banco de dados e redirecionar para o menu principal |
| 1. **Fluxos Alternativo**: |
| * 1. Seleciona a opção editar relatório |
| * 1. Filtra os relatórios por mês e/ou ano de referência |
| * 1. Escolhe um relatório dentre os mostrados na lista |
| * 1. Seleciona a opção avançar |
| * 1. Edita os dados da capa |
| * 1. Avançar |
| * 1. Edita os dados do cardápio |
| * 1. Avançar |
| * 1. Edita os dados da lista de ingredientes |
| * 1. Seleciona salvar para salvar as modificações no banco de dados |
| 1. **Fluxo de Exceção**: |
| * 1. Não existir relatório encontrado para o filtro informado |
| * + 1. A opção avançar não será habilitada |
| * 1. Usuário preenche algum campo incorretamente |
| * + 1. Aparecerá um pop-up informando que o campo está inválido e precisa ser verificado. |

|  |
| --- |
| **CASO DE USO 4** |
| 1. **Nome do caso de uso:** |
| Criar Relatório |
| 1. **Descrição resumida:** |
| usuário pode criar um relatório novo para a escola selecionada no contexto. |
| 1. **Ator:** Usuário |
| 1. **Atores Secundários:** |
| não Há |
| 1. **Pré-condição:** |
| * 1. Estar logado no sistema e ter a permissão para criar relatório |
| * 1. **Fluxo de eventos** |
| * + 1. Seleciona a opção criar relatório |
| * + 1. Informa o mês de referência |
| * + 1. Informa o ano de referência |
| * + 1. Preenche os dados da capa |
| * + 1. Avança |
| * + 1. Preenche os dados do cardápio |
| * + 1. Avança |
| * + 1. Preenche a lista de itens usados |
| * + 1. Seleciona salvar para salvar as modificações no banco de dados. |
| 1. **Pós-condições:** |
| Sistema deve inserir o novo relatório no banco de dados no banco de dados e redirecionar para o menu principal. |
| 1. **Fluxos Alternativo** |
| * 1. Usuário não informa o mês de referência |
| * 1. Ao clickar em avançar, aparece um pop-up alertando-o de que o campo está inválido ou não preenchido |
| * 1. Usuário não informa o ano de referência |
| * 1. Ao clickar em avançar, aparece um pop-up alertando-o de que o campo está inválido ou não preenchido. |
| 1. **Fluxo de Exceção** |
| * 1. Usuário cria um relatório com o mês e ano de referência igual ao de outro já existente no banco de dados para a mesma escola |
| * + 1. Aparece um pop-up informando que o relatório já existe e que ele deve ser editado e não criado novamente, impedindo o usuário de prosseguir. |

|  |
| --- |
| **CASO DE USO 5** |
| 1. **Nome do caso de uso:** |
| Cadastrar escola |
| 1. **Descrição resumida:** |
| O usuário pode cadastrar uma nova escola no banco de dados. |
| 1. **Ator:** |
| Usuário |
| 1. **Atores Secundários:** |
| Não há |
| 1. **Pré-condição:** |
| Estar logado no sistema e ter a permissão para cadastrar escola |
| 1. **Fluxo de eventos** |
| * 1. Seleciona “Gerenciar Escolas” |
| * 1. Seleciona a opção com um “+” |
| * 1. Preenche com os dados dessa nova escola |
| * 1. Salva |
| 1. **Pós-condições:** |
| Sistema deve redirecionar para a tela principal |
| 1. **Fluxos Alternativo**: |
| Não há |
| 1. **Fluxo de Exceção** |
| * 1. Usuário não tem permissão para cadastrar escola, mas tem permissão de gerenciar |
| * + 1. Aparece as demais opções mas a opção com o “+” é desabilitada |
| * 1. Usuário tenta inserir uma escola já existente no banco de dados |
| * + 1. O sistema bloqueia a operação informando que a escola já se encontra cadastrada. |

|  |
| --- |
| **CASO DE USO 6** |
| 1. **Nome do caso de uso:** |
| Editar Escola |
| 1. **Descrição resumida:** |
| O usuário pode editar uma escola do banco de dados. |
| 1. **Ator:** |
| Usuário |
| 1. **Atores Secundários:** |
| Não há |
| 1. **Pré-condição:** |
| Estar logado no sistema e ter a permissão para editar escola |
| 1. **Fluxo de eventos** |
| * 1. Seleciona “gerenciar escolas” |
| * 1. Seleciona uma escola para gerenciar no dropdown |
| * 1. Selecione a opção editar |
| * 1. Edita os campos da escola |
| * 1. Selecione “Salvar” |
| 1. **Pós-condições:** sistema deve atualizar o banco de dados e redirecionar para a tela principal |
| 1. **Fluxos Alternativo** |
| * 1. Cancelar edição |
| * + 1. Seleciona “gerenciar escolas” |
| * + 1. Seleciona uma escola para gerenciar no dropdown |
| * + 1. Selecione a opção editar |
| * + 1. Edita os campos da escola |
| * + 1. A opção voltar cancela toda edição feita e volta para tela anterior |
| 1. **Fluxo de Exceção** |
| * 1. Tentar mudar o inep (unico por escola) por um ja existente no banco |
| * 1. Edições desprezadas e operação cancelada. |
| * 1. Volta para tela de escolha de escola. |

|  |
| --- |
| **CASO DE USO 7** |
| 1. **Nome do caso de uso:** |
| Cadastrar usuário |
| 1. **Descrição resumida:** |
| O usuário pode cadastrar um novo usuário no banco de dados |
| 1. **Ator:** |
| Usuário |
| 1. **Atores Secundários:** |
| não há |
| 1. **Pré-condição:** |
| Estar logado no sistema como usuário administrador |
| 1. **Fluxo de eventos** |
| * 1. Seleciona “Gerenciar Usuários” |
| * 1. Seleciona “cadastrar usuário” |
| * 1. Preencha os campos |
| * 1. Selecione salvar |
| 1. **Pós-condições:** |
| Sistema deve criar o usuário no banco e redirecionar para a tela principal |
| 1. **Fluxos Alternativo**: |
| Não há |
| 1. **Fluxo de Exceção** |
| * 1. O usuário (username) já existir no banco |
| * + 1. pop-up dizendo que este usuário já existe no banco |
| * 1. Faltar preencher algum campo |
| * + 1. pop-up dizendo que tem campos não completos |

|  |
| --- |
| **CASO DE USO 8** |
| 1. **Nome do caso de uso:** |
| Editar usuário |
| 1. **Descrição resumida:** |
| Usuário pode editar os dados de algum usuário |
| 1. **Ator:** |
| Usuário |
| 1. **Atores Secundários:** |
| 1. **Pré-condição:** |
| Estar logado no sistema e ter permissão de edição de usuário |
| 1. **Fluxo de eventos** |
| * 1. Seleciona “Gerenciar Usuários” |
| * 1. Seleciona “Editar Usuario” |
| * 1. Preencha com o nome de usuario |
| * 1. Seleciona “Pesquisar” |
| * 1. Edite os dados apresentados |
| * 1. Seleciona “salvar” |
| 1. **Pós-condições:** sistema deve atualizar os dados do usuario e redirecionar para a tela principal. |
| 1. **Fluxos Alternativo** |
| * 1. Pesquisar por e-mail |
| * + 1. Seleciona “Editar Usuario” |
| * + 1. Preencha com o e-mail |
| * + 1. Seleciona “Pesquisar” |
| * + 1. Edite os dados apresentados |
| * + 1. Seleciona “salvar” |
| 1. **Fluxo de Exceção** |
| * 1. Nome de usuário não existente no banco |
| * + 1. Aparece um pop-up informando que não há nenhum usuário com o nome de usuário informado. |
| * + 1. permanece na tela para pesquisa de usuário |
| * 1. Email não existente no banco |
| * + 1. Aparece um pop-up informando que não há nenhum usuário com o email informado. |
| * + 1. permanece na tela para pesquisa de usuário |
| * 1. Nome usuário alterado para um já existente no banco |
| * + 1. Aparece um pop-up informando que o nome de usuário já existe no banco |
| * + 1. permanece na tela para edição. |
| * 1. Email alterado para um já existente no banco |
| * + 1. Aparece um pop-up informando que o email já existe no banco |
| * + 1. permanece na tela para edição. |

|  |
| --- |
| **CASO DE USO 9** |
| 1. **Nome do caso de uso:** |
| Excluir usuário |
| 1. **Descrição resumida:** |
| Usuário excluir outro usuário do banco |
| 1. **Ator:** |
| Usuário |
| 1. **Atores Secundários:** |
| Não há |
| 1. **Pré-condição:** |
| Estar logado no sistema e ter permissão de excluir usuário |
| 1. **Fluxo de eventos** |
| * 1. Seleciona “Gerenciar Usuários” |
| * 1. Seleciona “Excluir usuário” |
| * 1. Preencha com o nome de usuario |
| * 1. Seleciona “Excluir” |
| * 1. Selecione Ok no pop up de confirmação |
| 1. **Pós-condições:** |
| Sistema deve redirecionar para a tela principal |
| 1. **Fluxos Alternativo** |
| * 1. Excluir pelo e-mail |
| * + 1. Seleciona “Gerenciar Usuários” |
| * + 1. Seleciona “Excluir usuário” |
| * + 1. Preencha com o e-mail |
| * + 1. Seleciona “Excluir” |
| * + 1. Selecione Ok no pop up de confirmação |
| * 1. Usuário muda de ideia sobre a exclusão |
| * + 1. Seleciona “Gerenciar Usuários” |
| * + 1. Seleciona “Excluir usuário” |
| * + 1. Preencha com o e-mail |
| * + 1. Seleciona “Excluir” |
| * + 1. Seleciona a opção “cancelar” no pop-up de confirmação |
| 1. **Fluxo de Exceção** |
| * 1. Usuário tenta excluir o próprio usuário |
| * + 1. Aparece pop-up informando que a operação não é válida |
| * + 1. Operação é cancelada |
| * + 1. Redireciona para a pesquisa de usuários. |

|  |
| --- |
| **CASO DE USO 10** |
| 1. **Nome do caso de uso:** |
| Editar relatório corrente |
| 1. **Descrição resumida:** |
| Usuário pode de maneira mais rápida editar o último relatório feito pelo mesmo. |
| 1. **Ator:** |
| Usuário |
| 1. **Atores Secundários:** |
| Não há |
| 1. **Pré-condição:** |
| Estar logado no sistema e ter permissão para editar relatorio |
| 1. **Fluxo de eventos** |
| * 1. Seleciona a opção “Editar Relatório Corrente” |
| * 1. Edita os dados da capa |
| * 1. Avançar |
| * 1. Edita os dados do cardápio |
| * 1. Avançar |
| * 1. Edita os dados da lista de ingredientes |
| * 1. Seleciona a opção “salvar” para salvar as modificações no banco de dados. |
| 1. **Pós-condições:** |
| Sistema deve salvar as modificações feitas no relatório selecionado no banco de dados e redirecionar para o menu principal |
| 1. **Fluxos Alternativo**: |
| Não há |
| 1. **Fluxo de Exceção**: |
| * 1. Não existir relatório corrente |
| * + 1. A opção editar relatório corrente não será habilitada |
| * 1. Usuário preenche algum campo incorretamente |
| * + 1. Aparecerá um pop-up informando que o campo está inválido e precisa ser verificado. |

|  |
| --- |
| **CASO DE USO 11** |
| 1. **Nome do caso de uso:** |
| 1. **Descrição resumida:** |
| Usuário pode editar as permissões para cada um dos três tipos de usuário: Diretor, Administrador e Leitor |
| 1. **Ator:** |
| Usuário |
| 1. **Atores Secundários:** |
| Não há |
| 1. **Pré-condição:** |
| Estar logado no sistema e ter permissão para editar permissões |
| 1. **Fluxo de eventos** |
| * 1. Selecione a opção “Atualizar permissões” |
| * 1. Selecione o tipo de usuário dentre as três opções no dropDown |
| * 1. Marque as permissões que o mesmo passará a ter |
| * 1. Clique em atualizar |
| * 1. Selecione Ok no pop up de confirmação |
| 1. **Pós-condições:** |
| 1. Sistema deve redirecionar para a tela principal e atualizar as permissões no banco |
| 1. **Fluxos Alternativo** |
| * 1. Usuário desiste da atualização |
| * + 1. Selecione a opção “Atualizar permissões” |
| * + 1. Selecione o tipo de usuário dentre as três opções no dropDown |
| * + 1. Marque as permissões que o mesmo passará a ter |
| * + 1. Clique em atualizar |
| * + 1. Selecione cancelar no pop up de confirmação |
| 1. **Fluxo de Exceção**: |
| Não há |

|  |
| --- |
| **CASO DE USO 12** |
| 1. **Nome do caso de uso:** |
| Deslogar do sistema |
| 1. **Descrição resumida:** |
| usuário quer sair do seu acesso do sistema e voltar a tela de login |
| 1. **Ator:** |
| Usuário |
| 1. **Atores Secundários:** |
| não há |
| 1. **Pré-condição:** |
| Estar logado no sistema |
| 1. **Fluxo de eventos** |
| * 1. Selecione a opção “sair” |
| 1. **Pós-condições:** |
| sistema deve redirecionar para a tela de login |
| 1. **Fluxos Alternativo**: não há |
| 1. **Fluxo de Exceção**: não há |

|  |
| --- |
| **CASO DE USO 13** |
| 1. **Nome do caso de uso:** |
| Mudar contexto da escola |
| 1. **Descrição resumida:** |
| Usuário deseja trocar o contexto da escola na qual está trabalhando para uma outra escola |
| 1. **Ator:** |
| Usuário |
| 1. **Atores Secundários:** |
| Não há |
| 1. **Pré-condição:** |
| Estar logado no sistema |
| 1. **Fluxo de eventos** |
| * 1. Selecione a opção “Alterar Escola” |
| * 1. Selecione a escola desejada |
| * 1. Selecione trocar |
| 1. **Pós-condições:** |
| Sistema deve redirecionar para a tela principal com o contexto atualizado |
| 1. **Fluxos Alternativo**: |
| * 1. Desiste da operação |
| * + 1. Selecione a opção “Voltar” |
| 1. **Fluxo de Exceção**: |
| Não há |

**ANEXO IV**

CASOS DE TESTE

Caso de teste UC001 – CT 001

1. **Identificação do cenário:** Fluxo básico
2. **Pré Memória**
   1. **Massa de dados**
      1. Usuario existente na tabela “usuário” do banco de dados
      2. Senha S1 do usuário
   2. **Propriedades**
      1. Não há
3. **Pós Memória**
   1. **Massa de dados:**
      1. Tela principal.usuário = usuário
   2. **Propriedades**:
      1. Logado no sistema
4. **Entrada e Saída Esperados**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Passo | Descrição | Variáveis | | | |
| Var1 | Val1 | Var2 | Val2 |
| 1 | Preencher campo usuário | userv | user |  |  |
| 2 | Preencher campo senha |  |  | senha | S1 |
| 3 | Clicar em “logar” |  |  |  |  |

**ANEXO V**

TESTES DOCUMENTADOS

|  |  |  |
| --- | --- | --- |
| Tipo de Teste | Classe/Caso de uso | Resultado |
| Unitário | Calendário |  |
| Unitário | Escola |  |
| Unitário | MaisEducacao |  |
| Unitário | Permissoes |  |
| Unitário | RefeicoesDados |  |
| Unitário | TelaCadastro |  |
| Unitário | TelaLogin |  |
| Unitário | TiposDeUsuario |  |
| Unitário | Usuario |  |