ESCOLA SENAI

“PROF. DR. EURYCLIDES DE JESUS ZERBINI”

Flavio Freire De Nardi

Otávio Augusto da Silva

Murillo Fornazza Costa

Vitor Moretti Borin

**MARKET FOV**

Campinas SP

2024

Flavio Freire De Nardi

Otávio Augusto da Silva

Murillo Fornazza Costa

Vitor Moretti Borin

**MARKET FOV**

Projeto apresentado à Escola SENAI “Prof. Dr. Euryclides de Jesus Zerbini” para obtenção do certificado de conclusão do Curso Técnico de Desenvolvimento de Sistemas.

Orientador: Renato de Mattos Onofre, Paulo Henrique Pansani

Campinas SP

2024

Flavio Freire De Nardi

Otávio Augusto da Silva

Murillo Fornazza Costa

Vitor Moretti Borin

**MARKET FOV**

Trabalho de conclusão de curso aprovado como requisito parcial para obtenção do grau de técnico, do curso Técnico de Desenvolvimento de Sistemas da Escola SENAI “Prof. Dr. Euryclides de Jesus Zerbini”.

**BANCA EXAMINADORA**

1º Examinador

2º Examinador

3º Examinador

Campinas 2024

**AGRADECIMENTOS**

Gostaríamos de expressar nossa profunda gratidão pelo impacto positivo que os professores Renato e Paulo tiveram em nossa jornada acadêmica e profissional no desenvolvimento de sistemas. Os conhecimentos obtidos foram de grande importância para nos tornar desenvolvedores de verdade.

Renato, o senhor nos ensinou e nos cobrou desde o início do curso para que cheguemos até aqui com o conhecimento necessário para passarmos por essa última etapa dessa jornada, se não fosse essa forte cobrança nos pequenos detalhes dos sistemas acho que não iriamos ter a confiança que temos para passar, fez toda a diferença.

Paulo, aprendemos muito com o senhor em diversos aspectos, por mais que as vezes algumas reuniões não planejadas ou palestras tenham atrapalhados principalmente as suas aulas porque sempre ocorriam no meio delas, seus ensinamentos nos ajudaram muito e agora temos uma confiança maior para finalizarmos o curso.

Agradecemos não apenas pelo conhecimento transmitido, mas também pelo incentivo e pela confiança depositada em nós. Vocês não foram apenas professores, mas verdadeiros mentores que ajudaram a moldar nossa carreira no mundo da tecnologia em geral, sempre levaremos os ensinamentos passados não só por vocês mais por todos os professores que também nos auxiliaram.

Esperamos poder aplicar tudo o que aprendemos com vocês e fazer valer todo o esforço e dedicação que investiram em nosso crescimento. Somos gratos por cada aula, por cada conselho e por cada momento de aprendizado.

Muito obrigado por tudo!

Atenciosamente,

MaxFOV

**EPÍGRAFE**

**“*A vida não tem sentido;***

***Cabe a nós dar-lhe algum, ou não*.”**

**- Albert Camus**

**SUMÁRIO**

[1 INTRODUÇÃO 7](#_Toc181905836)

[2 JUSTIFICATIVA 8](#_Toc181905837)

[3 OBJETIVOS 9](#_Toc181905838)

[3.1. Objetivos Gerais 9](#_Toc181905839)

[3.2. Objetivos Específicos 9](#_Toc181905840)

[4 PRODUCT BACKLOG 10](#_Toc181905841)

[5 REQUISITOS NÃO FUNCIONAIS 10](#_Toc181905842)

[6 PREMISSAS 11](#_Toc181905843)

[7 RESTRIÇÕES 12](#_Toc181905844)

[8 ANÁLISE DE RISCOS DE UM PROJETO 13](#_Toc181905845)

[8.1. Nível e Planos de Ação para os Riscos 13](#_Toc181905846)

[9 SPRINTS 14](#_Toc181905847)

[9.1. Primeiro Sprint 14](#_Toc181905848)

[9.1.1. Product Backlog 14](#_Toc181905849)

[9.1.2. Sprint Backlog 14](#_Toc181905850)

[9.1.3. Burn Down Chart 15](#_Toc181905851)

[9.1.4. Diagramas 15](#_Toc181905852)

[9.1.5. Plano de testes 15](#_Toc181905853)

[9.1.5.1. Resultados 15](#_Toc181905854)

[9.1.6. Kanban e Retrospectiva 15](#_Toc181905855)

[9.2. Segundo Sprint 16](#_Toc181905856)

[9.2.1. Product Backlog 16](#_Toc181905857)

[9.2.2. Sprint Backlog 16](#_Toc181905858)

[9.2.3. Burn Down Chart 17](#_Toc181905859)

[9.2.4. Diagramas 18](#_Toc181905860)

[**NARRATIVA DE CASO DE USO** 18](#_Toc181905861)

[9.2.5. Plano de testes 23](#_Toc181905862)

[9.2.6. Resultados 23](#_Toc181905863)

[9.2.7. Kanban e Retrospectiva 24](#_Toc181905864)

[10. Modelo de Dados 25](#_Toc181905867)

[10.1. Diagrama de Entidade e Relacionamento 25](#_Toc181905868)

[10.2. Modelo lógico do banco de dados 26](#_Toc181905869)

[10.3. Dicionário de dados 27](#_Toc181905870)

[11. PRINCIPAIS TELAS DO SISTEMA 30](#_Toc181905871)

[12. CONCLUSÃO 32](#_Toc181905872)

[12.2. Escreva os resultados obtidos 32](#_Toc181905873)

[12.3. Constatações 32](#_Toc181905874)

[12.4. Sugestões de possíveis aperfeiçoamentos técnicos 32](#_Toc181905875)

[13. REFERÊNCIAS 33](#_Toc181905876)

[14. GLOSSÁRIO 34](#_Toc181905877)

[15. ANEXOS 35](#_Toc181905878)

# INTRODUÇÃO

A transformação digital tem impactado profundamente diversos setores, e o mercado varejista não é exceção. A modernização de sistemas voltados para a gestão de mercados pode melhorar a eficiência e acessibilidade no comércio, especialmente para pequenos e médios empresários que enfrentam desafios como a alta concorrência e a necessidade de otimização dos processos. Em resposta a esses desafios, o MarketFOV foi desenvolvido com o objetivo de oferecer uma plataforma completa e acessível para donos de mercados que buscam simplificar e aprimorar a gestão de seus negócios.

O MarketFOV é uma solução de sistema de mercado que vai além do cadastro e compra de produtos. Ele permite que esses processos sejam realizados de forma prática e segura pela web, com uma interface intuitiva e acessível, projetada para facilitar o uso e otimizar a experiência dos usuários. Além disso, o sistema incorpora funcionalidades semelhantes às de um ERP (Enterprise Resource Planning), permitindo a visualização de relatórios de vendas por filtros de data específicos, o que auxilia os proprietários a tomar decisões informadas e estratégicas com base nos dados de desempenho do seu negócio.

Inspirado pela necessidade de democratizar o acesso a tecnologias de gestão, o MarketFOV busca capacitar pequenos e médios empresários com ferramentas práticas e modernas. Assim como o legado de grandes contribuições na área educacional, como a de pensadores que promoveram a universalização do acesso ao conhecimento, nosso projeto pretende transformar o mercado varejista, promovendo inclusão e eficiência.

# JUSTIFICATIVA

Com o avanço da tecnologia e a crescente digitalização do comércio, os processos de gestão de pequenos e médios mercados enfrentam novos desafios e exigências. A utilização de sistemas tradicionais, muitas vezes lentos e pouco eficientes, impede o pleno desenvolvimento desses estabelecimentos, que acabam enfrentando dificuldades para gerenciar estoques, realizar vendas e monitorar o desempenho de seus negócios. Diante dessa realidade, a equipe MaxFOV desenvolveu o MarketFOV, uma solução moderna que combina simplicidade, rapidez e eficiência, proporcionando um sistema de mercado voltado para as necessidades atuais de praticidade e controle.

O objetivo principal do MarketFOV é otimizar a gestão de produtos e facilitar os processos cotidianos de mercado, como o cadastro, atualização e exclusão de itens de forma simples e intuitiva. Baseado em princípios de ERP, o sistema também oferece funcionalidades para o monitoramento de vendas e a geração de relatórios filtrados por data, permitindo que os proprietários tenham uma visão clara e precisa do desempenho do seu negócio. Essa capacidade de controle e organização resolve as principais queixas de nossos entrevistados em pesquisa de campo: a lentidão e travamentos frequentes, além do descontrole no gerenciamento de produtos.

Com o MarketFOV, a equipe MaxFOV busca ir além da simples automação. Pretendemos criar uma solução que democratize o acesso a uma gestão mais eficiente e moderna, permitindo que pequenos e médios empresários se beneficiem de um sistema robusto e acessível. Nossa plataforma visa transformar o gerenciamento de mercados em uma experiência ágil e satisfatória, onde cada transação e atualização seja feita com facilidade e velocidade. Dessa forma, o MarketFOV se posiciona não apenas como um sistema de mercado, mas como uma ferramenta estratégica que permite ao usuário melhorar continuamente sua gestão e acompanhar as demandas e tendências atuais do comércio.

# OBJETIVOS

O objetivo do projeto Market FOV é tornar o gerenciamento de um mercado mais simples e rápido com um sistema universal para todos os pequenos mercados que não tem o orçamento para um sistema próprio.

# Objetivos Gerais

O Market FOV visa estabelecer um padrão fácil de compreender para o gerenciamento do seu mercado, sendo capaz de gerar uma retrospectiva de vendas diárias, semanais e mensais. O sistema também deve permitir o usuário realizar vendas e ver sua retrospectiva, enquanto o gerente deve ser capaz de cadastrar o mercado e novos produtos, assim como apagá-los do estoque e alterar dados do mercado e/ou de produtos caso necessário.

# Objetivos Específicos

Para a realização desse projeto foram utilizadas algumas ferramentas, dentre elas, utilizamos HTML, CSS e JavaScript para desenvolvimento do visual do site. Também foi utilizado PHP e JavaScript novamente para desenvolvimento das funcionalidades do site. Quanto em relação à armazenamento de dados, recorremos ao banco de dados MySQL. Por último, o versionamento do código foi controlado pelo GitHub.

# PRODUCT BACKLOG

MFB1 – Realizar cadastro e login de mercado

MFB2 – Realizar cadastro e busca de produtos no estoque

MFB3 – Buscar dados sobre compras feitas

MFB4 – Alterar dados de produtos já cadastrados

MFB5 – Alterar dados do mercado após seu cadastro

MFB6 – Deletar produtos que acabaram/não estão mais em estoque

MFB7 – Buscar dados do relatório de vendas

MFB8 – Realizar vendas

# REQUISITOS NÃO FUNCIONAIS

RNF1 – O design deve ser simples e intuitivo para facilitar o gerenciamento.

RNF2 – O site deve suportar até 5 usuários simultâneos.

RNF3 – O site deve realizar as buscas de dados de forma rápida e eficiente.

RNF4 – Assim como as buscas de dados, a utilização geral do site também deve funcionar de maneira rápida e eficiente.

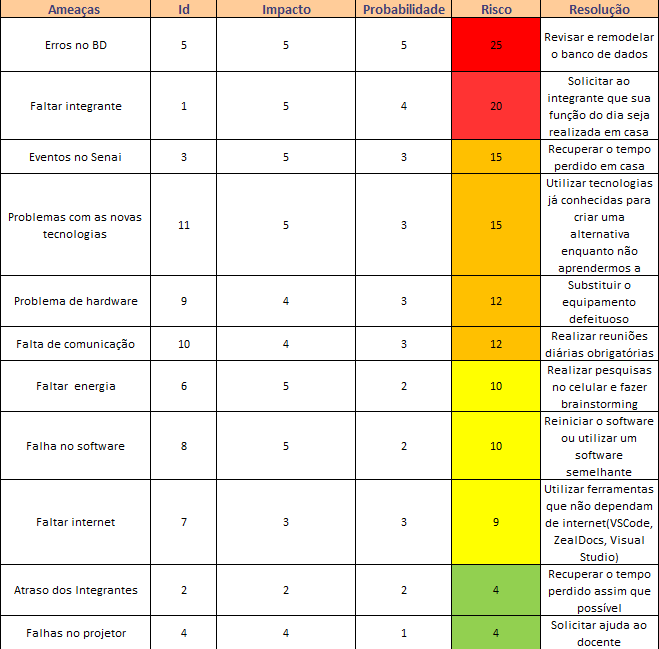
# PREMISSAS

* O sistema será desenvolvido utilizando tecnologias web amplamente aceitas e suportadas.
* O cliente fornecerá as especificações detalhadas sobre os produtos e operações do mercado até a data de início do desenvolvimento.
* A equipe de desenvolvimento terá acesso contínuo a feedback dos usuários finais para garantir que o sistema atenda às necessidades reais.
* O ambiente de teste será disponibilizado antes do início da fase de implementação para garantir a validação das funcionalidades.
* O sistema será implementado em uma infraestrutura que suporta a carga máxima prevista de usuários simultâneos, sem comprometer a performance.
* Todos os dados necessários para o cadastro de produtos e mercados serão fornecidos pelo cliente em formato digital.
* O projeto terá um cronograma que permitirá a realização de revisões regulares com as partes interessadas para garantir a aderência aos requisitos.

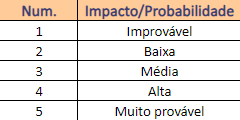
# RESTRIÇÕES

* O ambiente de trabalho não está disponível aos finais de semana, pois nem todos os integrantes tem como trabalhar de casa.
* Somente serão utilizados softwares livres para o desenvolvimento da aplicação.
* O projeto deve ser concluído dentro do prazo estabelecido, que é de seis meses a partir da data de início do desenvolvimento.
* A equipe deve utilizar apenas ferramentas de desenvolvimento e bibliotecas que estejam em conformidade com as políticas de licenciamento do cliente.
* O acesso ao banco de dados deve ser limitado a um número específico de usuários para garantir a segurança das informações.
* O sistema deve ser compatível com os navegadores mais utilizados, sem suporte para navegadores desatualizados.
* A documentação do projeto deve ser mantida atualizada e disponível para consulta durante todo o ciclo de vida do desenvolvimento.

# ANÁLISE DE RISCOS DE UM PROJETO



# Nível e Planos de Ação para os Riscos



# SPRINTS

# Primeiro Sprint

Nosso primeiro Sprint foi focado principalmente em pensar no tema do nosso projeto assim como realizar o design do nosso projeto junto de grande parte do nosso front-end.

Nossa meta nesse Sprint foi entregar o front-end e o banco de dados já pronto para podermos dar inicio no back-end que seria o nosso foco principal do segundo Sprint.

# Product Backlog

MFB1 – Realizar cadastro e login de mercado

MFB2 – Realizar cadastro e busca de produtos no estoque

MFB3 – Buscar dados sobre compras feitas

MFB4 – Alterar dados de produtos já cadastrados

MFB5 – Alterar dados do mercado após seu cadastro

MFB6 – Deletar produtos que acabaram/não estão mais em estoque

MFB7 – Buscar dados do relatório de vendas

MFB8 – Realizar vendas

# Sprint Backlog

MS1B1 – Pensar no design das telas

MS1B2 – Fazer as telas

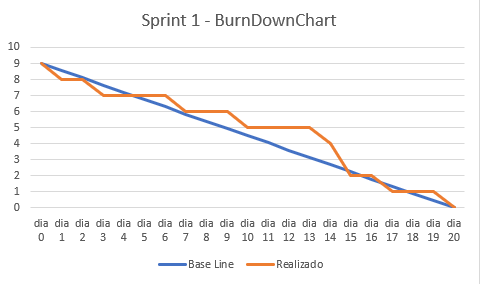
MS1B3 – Criar o MER do banco de dados

MS1B4 – Criar o modelo lógico do banco de dados

MS1B5 – Criar o modelo físico do banco de dados

MS1B6 – Fazer a documentação do projeto

# Burn Down Chart



# Diagramas

[Apresenta](http://www.devmedia.com.br/curso/introducao-a-uml/128) os principais diagramas realizados.

# Plano de testes

O plano de teste é uma maneira de encontrar defeitos e bugs no sistema para serem futuramente corrigidos. Mesmo que durante a programação sempre foram realizados alguns, ainda sim no plano são realizados testes finais passo a passo.

# Resultados

Resultados apresentados nos planos de testes

# Kanban e Retrospectiva

[Apresenta](http://www.devmedia.com.br/curso/introducao-a-uml/128) imagens do quadro de Kanban e as conclusões da retrospectiva

# 9.2. Segundo Sprint

No Nosso segundo Sprint, nós focamos em fazer o Back-End, criando as APIs em JavaScript e as conexões com o banco de dados criado anteriormente usando PHP, também fizemos as validações dos campos.

Nosso objetivo nesse sprint foi entregar um Back-End e uma API funcional, que faça um CRUD(Crie, Busque, Atualize e Delete) com o banco de dados de forma correta.

# 9.2.1. Product Backlog

MFB1 – Realizar cadastro e login de mercado

MFB2 – Realizar cadastro e busca de produtos no estoque

MFB3 – Buscar dados sobre compras feitas

MFB4 – Alterar dados de produtos já cadastrados

MFB5 – Alterar dados do mercado após seu cadastro

MFB6 – Deletar produtos que acabaram/não estão mais em estoque

MFB7 – Buscar dados do relatório de vendas

MFB8 – Realizar vendas

# 9.2.2. Sprint Backlog

MS2B1 - Puxar dados de relatório de venda

MS2B2 - Método para alterar cadastro de mercado

MS2B3 - Método para deletar produtos

MS2B3 - Método para alterar produtos cadastrados

MS2B4 - Buscar produtos no banco de dados

MS2B5 - Buscar compras feitas

MS2B6 - Fazer login de mecado

MS2B7 - Fazer validação dos campos

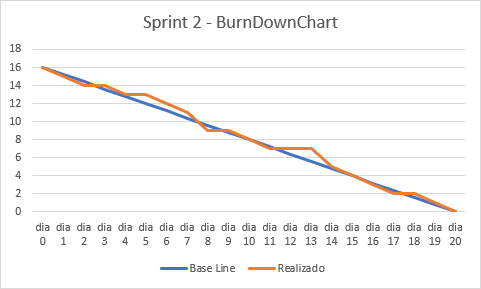
MS2B8 - Fazer cadastro de produtos no estoque

MS2B9 - Fazer cadastro de vendas

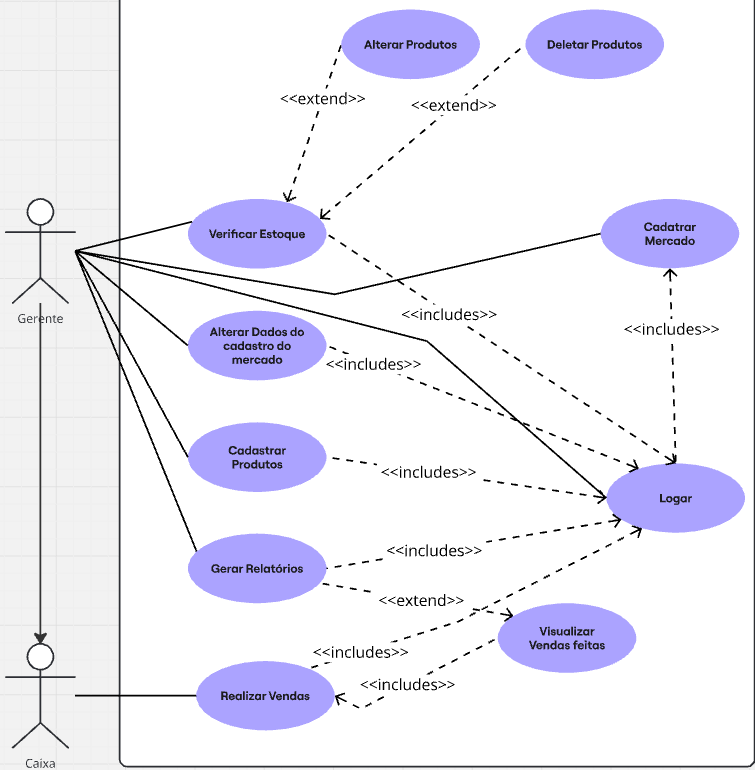
MS2B10 - Fazer cadastro de mercado

MS2B11 – Fazer a documentação

# 9.2.3. Burn Down Chart



# Diagramas



|  |
| --- |
| **NARRATIVA DE CASO DE USO** |
| Sigla/Nome: CSU01 - Verificar Estoque Objetivo: Permitir que o gerente consulte e visualize a quantidade de produtos disponíveis no estoque. Frequência estimada: Diariamente. Ator Principal: Gerente |
| **Cenário Principal** |
| 1. O caso de uso inicia quando o Gerente acessa o sistema e seleciona a opção Verificar Estoque no menu principal. |
| 1. O sistema apresenta uma tela contendo as informações do estoque, com a quantidade de produtos disponíveis e opções de filtragem. |
| 1. O Gerente pode realizar consultas específicas, como busca por produto ou categoria. |
| 1. O sistema retorna os resultados da consulta solicitada. |
| 1. O Gerente visualiza os dados do estoque e, se necessário, pode tomar decisões operacionais (como reposição ou exclusão de produtos). |
| **Cenário Alternativo** |
| 3.1 - Consulta detalhada de produto específico |
| * 3.1.1 O Gerente insere o nome ou código do produto na caixa de busca. |
| * 3.1.2 O sistema realiza uma busca e apresenta as informações detalhadas sobre o produto, como quantidade disponível, histórico de reposições e data de validade. |
| * 3.1.3 O Gerente visualiza as informações e pressiona Voltar para retornar à tela principal. |
| 3.2 - Produto indisponível no estoque |
| * 3.2.1 O Gerente realiza uma consulta por um produto e o sistema informa que o produto está fora de estoque. |
| * 3.2.2 O sistema sugere ações, como reposição ou exclusão, e o Gerente toma as medidas necessárias. |
| * 3.2.3 O Gerente pressiona Voltar para retornar à tela principal. |
| **Cenários de Exceção** |
| 4.1 Falha na busca por produto |
| * Caso o Gerente insira um nome ou código de produto que não exista, o sistema exibe uma mensagem de erro e retorna ao passo 2. |
| **Pré-Condições** |
| 1. O Gerente deve estar autenticado no sistema (CSU03 – Logar). |
| 1. O sistema deve estar atualizado com as informações mais recentes do estoque. |
| **Pós-Condições** |
| 1. O estoque será visualizado corretamente. |
| 1. Se aplicável, um registro de consulta será gravado no log do sistema, com data e hora. |
| Sigla/Nome: CSU02 - Cadastrar Produtos Objetivo: Permitir que o gerente cadastre novos produtos no sistema. Frequência estimada: Variável, dependendo da entrada de novos produtos. Ator Principal: Gerente |
| **Cenário Principal** |
| 1. O caso de uso inicia quando o Gerente acessa o sistema e seleciona a opção Cadastrar Produtos no menu. |
| 1. O sistema apresenta uma tela com campos a serem preenchidos, como nome, descrição, quantidade em estoque, preço e código do produto. |
| 1. O Gerente preenche os dados e pressiona o botão Salvar. |
| 1. O sistema valida as informações fornecidas. |
| 1. Após validação, o sistema salva os dados do novo produto no banco de dados e confirma a inclusão. |
| 1. O Gerente pode cadastrar outros produtos ou voltar ao menu anterior. |
| **Cenário Alternativo** |
| 3.1 - Produto já cadastrado |
| * 3.1.1 Se o Gerente inserir um código de produto já existente, o sistema informa que o produto já está cadastrado e retorna ao passo 2. |
| **Cenários de Exceção** |
| 4.1 - Falha na validação dos dados |
| * Caso algum campo obrigatório não seja preenchido corretamente, o sistema informa o erro e retorna ao passo 2 para correção. |
| Pré-Condições |
| 1. O Gerente deve estar autenticado no sistema (CSU03 – Logar). |
| **Pós-Condições** |
| 1. O novo produto será registrado no sistema. |
| 1. Um registro será gerado no log, indicando a data e o responsável pelo cadastro. |
|  |
| Sigla/Nome: CSU03 - Logar Objetivo: Controlar o acesso ao sistema. Frequência estimada: 10 vezes ao dia, sempre que necessário acessar o sistema. Ator Principal: Gerente e Caixa |
| **Cenário Principal** |
| 1. O caso de uso inicia quando o Gerente ou Caixa acessa o sistema e seleciona a opção Logar. |
| 1. O sistema apresenta uma tela contendo campos de login e senha. |
| 1. O usuário insere o login e a senha. |
| 1. O sistema valida as informações e, se estiverem corretas, abre a tela principal. |
| 1. O usuário pode realizar as operações disponíveis no sistema. |
| **Cenários de Exceção** |
| 4.1 - Falha na validação do login ou senha |
| * Se o login ou senha estiverem incorretos, o sistema informa o erro e retorna ao passo 2, permitindo novas tentativas. |
| **Pré-Condições** |
| 1. O usuário deve estar cadastrado no sistema. |
| **Pós-Condições** |
| 1. Um registro no arquivo de log será gerado, contendo a data, hora e nome do usuário que acessou o sistema. |
|  |
| Sigla/Nome: CSU04 - Realizar Vendas Objetivo: Registrar as vendas realizadas no sistema. Frequência estimada: Alta, várias vezes ao dia. Ator Principal: Caixa |
| **Cenário Principal** |
| 1. O caso de uso inicia quando o Caixa acessa o menu Vendas e seleciona a opção Realizar Venda. |
| 1. O sistema apresenta uma tela onde o Caixa deve preencher os dados do cliente (se houver), selecionar os produtos vendidos e inserir a quantidade. |
| 1. O sistema calcula o valor total da venda e exibe as opções de pagamento. |
| 1. O Caixa seleciona a forma de pagamento. |
| 1. O sistema finaliza a venda e atualiza o estoque. |
| 1. O Caixa pode realizar outra venda ou voltar ao menu principal. |
| **Cenário Alternativo** |
| 4.1 - Forma de pagamento em dinheiro |
| * O Caixa seleciona a opção Dinheiro e insere o valor recebido. |
| * O sistema calcula o troco, se houver, e finaliza a venda. |
| 4.2 - Forma de pagamento por cartão |
| * O Caixa seleciona a opção Cartão de Crédito/Débito e realiza a leitura dos dados do cartão. |
| * O sistema processa o pagamento e, se autorizado, finaliza a venda. |
| **Cenários de Exceção** |
| 4.3 - Produto não disponível em estoque |
| * Caso algum dos produtos selecionados esteja indisponível, o sistema informa o erro e retorna ao passo 2 para ajustes na venda. |
| **Pré-Condições** |
| 1. O Caixa deve estar autenticado no sistema (CSU03 – Logar). |
| 1. O caixa deve estar aberto. |
| **Pós-Condições** |
| 1. A venda será registrada no sistema. |
| 1. O estoque será atualizado. |
| 1. Um registro no arquivo de log será gerado contendo os detalhes da venda. |
|  |
| Sigla/Nome: CSU05 - Gerar Relatórios Objetivo: Permitir que o gerente gere relatórios de vendas, produtos e estoque. Frequência estimada: Diariamente. Ator Principal: Gerente |
| **Cenário Principal** |
| 1. O caso de uso inicia quando o Gerente acessa o menu Relatórios e seleciona a opção Gerar Relatórios. |
| 1. O sistema apresenta uma tela onde o Gerente pode selecionar o tipo de relatório (vendas, produtos ou estoque). |
| 1. O Gerente escolhe o tipo de relatório e define os filtros necessários (período, produto, etc.). |
| 1. O sistema gera o relatório solicitado. |
| 1. O Gerente visualiza ou exporta o relatório gerado. |
| **Cenário Alternativo** |
| 3.1 - Gerar relatório detalhado por produto |
| * O Gerente seleciona o relatório de Produtos e aplica filtros para detalhar a consulta por produto específico ou categoria. |
| * O sistema gera o relatório conforme os filtros aplicados. |
| **Cenários de Exceção** |
| 4.1 - Falha na geração do relatório |
| * Caso ocorra algum erro durante a geração do relatório, o sistema informa o erro e permite que o Gerente tente novamente. |
| **Pré-Condições** |
| 1. O Gerente deve estar autenticado no sistema (CSU03 – Logar). |
| **Pós-Condições** |
| 1. O relatório será gerado e poderá ser visualizado ou exportado. |
| 1. Um registro será gerado no log contendo as informações do relatório gerado. |

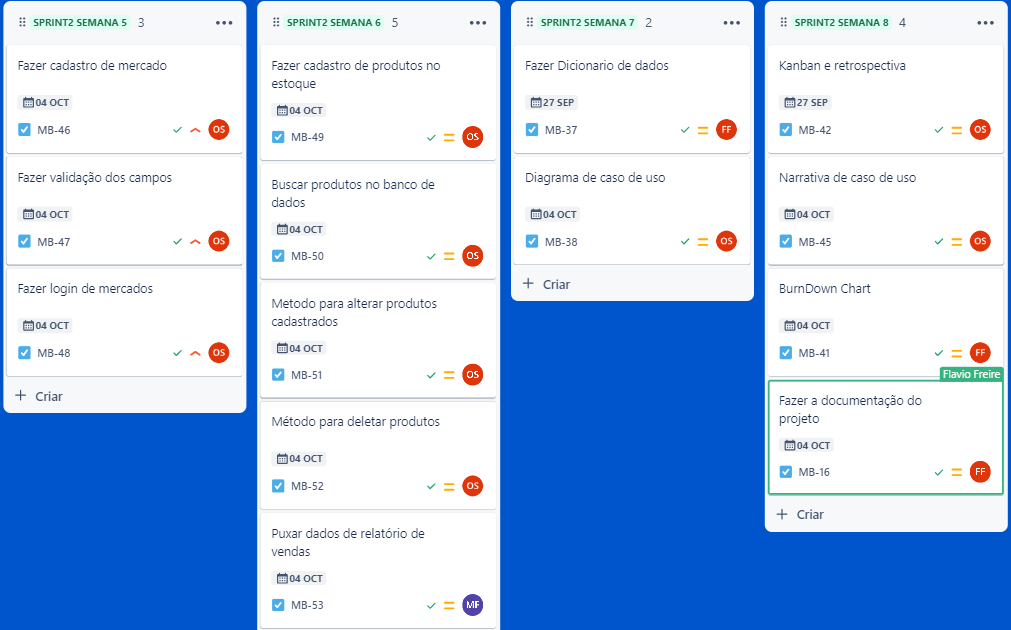
# Plano de testes

O plano de teste é uma maneira de encontrar defeitos e bugs no sistema para serem futuramente corrigidos. Mesmo que durante a programação sempre foram realizados alguns, ainda sim no plano são realizados testes finais passo a passo.

# Resultados

Resultados apresentados nos planos de testes

# Kanban e Retrospectiva

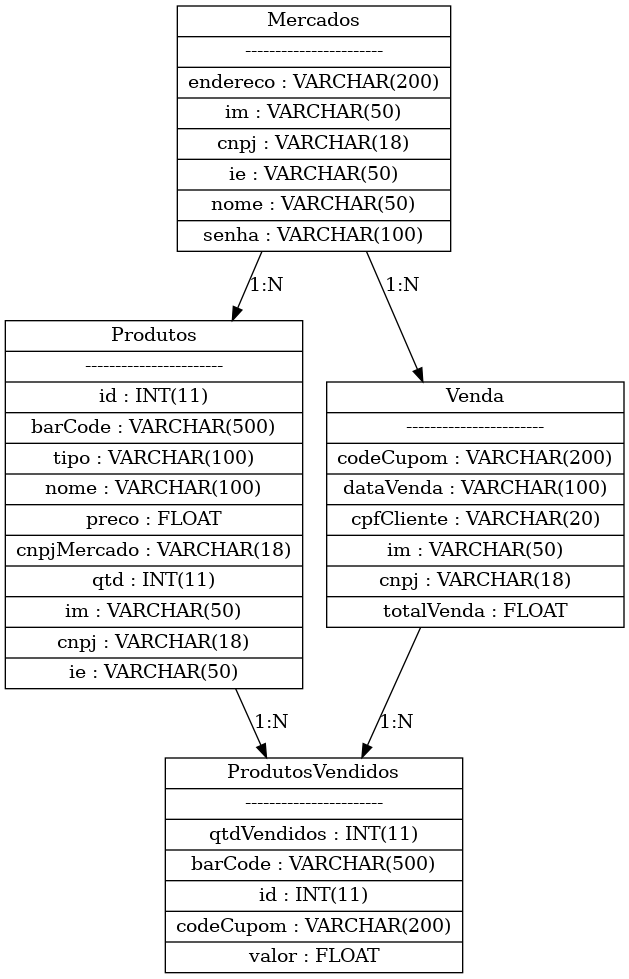




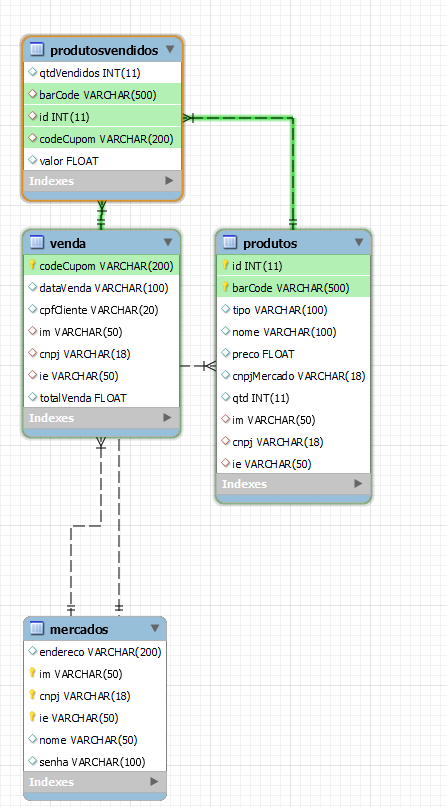


# Modelo de Dados

# Diagrama de Entidade e Relacionamento



# Modelo lógico do banco de dados



# Dicionário de dados

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Tabela** | Mercados | | | | |
| **Descrição** | Armazena as informações dos mercados | | | | |
| **Observações** | - | | | | |
| **Campos** | | | | | |
| **Nome** | | **Descrição** | **Tipo de dados** | **Tamanho** | **Restrições de domínio** |
| **Endereco** | | Localização do mercado | Varchar | 200 | Not null |
| **IM** | | Inscrição municipal do mercado | Varchar | 50 | PK / Not null |
| **CNPJ** | | CNPJ do mercado | Varchar | 18 | PK / Not null |
| **IE** | | Inscrição estadual do mercado | Varchar | 50 | PK / Not null |
| **Nome** | | Nome do mercado | Varchar | 50 | Not null |
| **Senha** | | Senha do mercado | Varchar | 100 | Not null |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tabela** | Produtos | | | | |
| **Descrição** | Armazena informações dos produtos cadastrados | | | | |
| **Observações** | Tem três foreign Keys que referenciam a tabela Mercados | | | | |
| **Campos** | | | | | |
| **Nome** | | **Descrição** | **Tipo de dados** | **Tamanho** | **Restrições de domínio** |
| **ID** | | Id do produto | Int | 11 | AI / PK/ Not null |
| **Barcode** | | Código de barra do produto | Varchar | 500 | PK / Not null |
| **Tipo** | | Tipo de produto | Varchar | 100 | Not null |
| **Nome** | | Nome do produto | Varchar | 100 | Not null |
| **Preco** | | Preço do produto | float | - | Not null |
| **Qtd** | | Quantidade de produtos em estoque | Int | 11 | Not null |
| **IM** | | Foreign key do IM do mercado | Varchar | 50 | FK / Not null |
| **CNPJ** | | Foreign key do CNPJ do mercado | Varchar | 18 | FK / Not null |
| **IE** | | Foreign key do IE do mercado | Varchar | 50 | FK / Not null |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tabela** | Produtosvendidos | | | | |
| **Descrição** | Armazena as informações de produtos que foram vendidos | | | | |
| **Observações** | Referenciam as tabelas Venda e Produto em três foreign keys | | | | |
| **Campos** | | | | | |
| **Nome** | | **Descrição** | **Tipo de dados** | **Tamanho** | **Restrições de domínio** |
| **QtdVendidos** | | Quatidade de produtos que foram vendidos | Int | 11 | Not null |
| **BarCode** | | Foreign key do código de barras do prod. | Varchar | 500 | Not null |
| **ID** | | Foreign key do ID do produto | Int | 11 | Not null |
| **CodeCupom** | | Foreign key do código de cupom da tabela de venda | Varchar | 50 | Not null |
| **Valor** | | Valor do produto vendido | float | - | Not null |

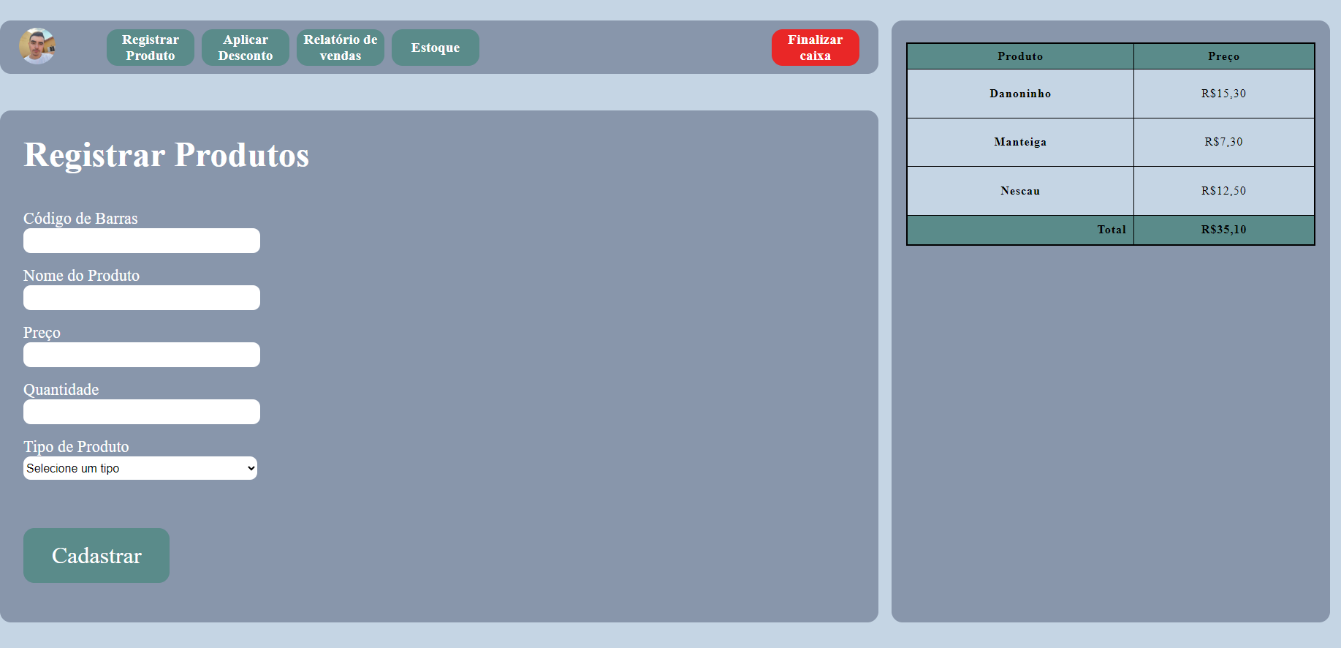
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tabela** | Venda | | | | |
| **Descrição** | Armazena as informações das vendas | | | | |
| **Observações** | Possui três foreign Keys que referenciam a tabela Mercado | | | | |
| **Campos** | | | | | |
| **Nome** | | **Descrição** | **Tipo de dados** | **Tamanho** | **Restrições de domínio** |
| **CodeCupom** | | Armazena o código de um cupom | Varchar | 200 | PK / Not null |
| **DataVenda** | | Data da venda | Varchar | 100 | Not null |
| **CpfCliente** | | CPF do cliente | Varchar | 20 | Not null |
| **IM** | | Foreign key da inscrição municipal do mercado | Varchar | 50 | FK / Not null |
| **CNPJ** | | Foreign key do CNPJ do mercado | Varchar | 18 | FK / Not null |
| **IE** | | Foreign key da inscrição estadual do mercado | Varchar | 50 | FK / Not null |
| **TotalVenda** | | Valor total da venda | Float | - | Not null |

# PRINCIPAIS TELAS DO SISTEMA

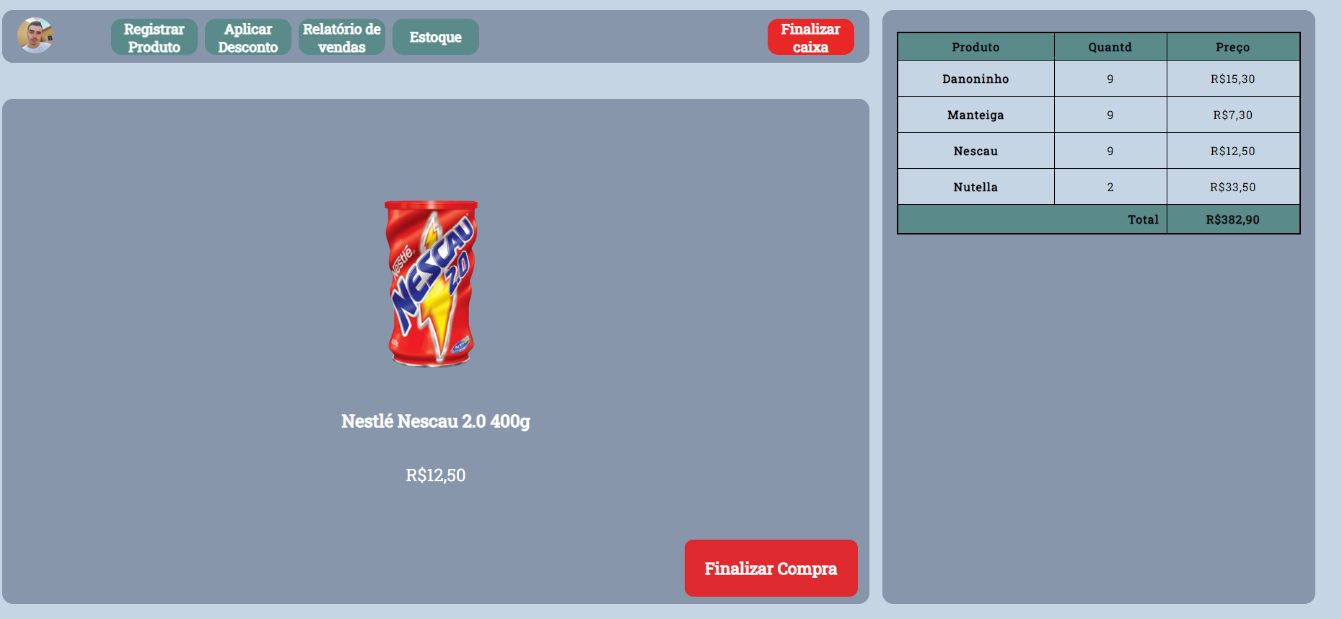


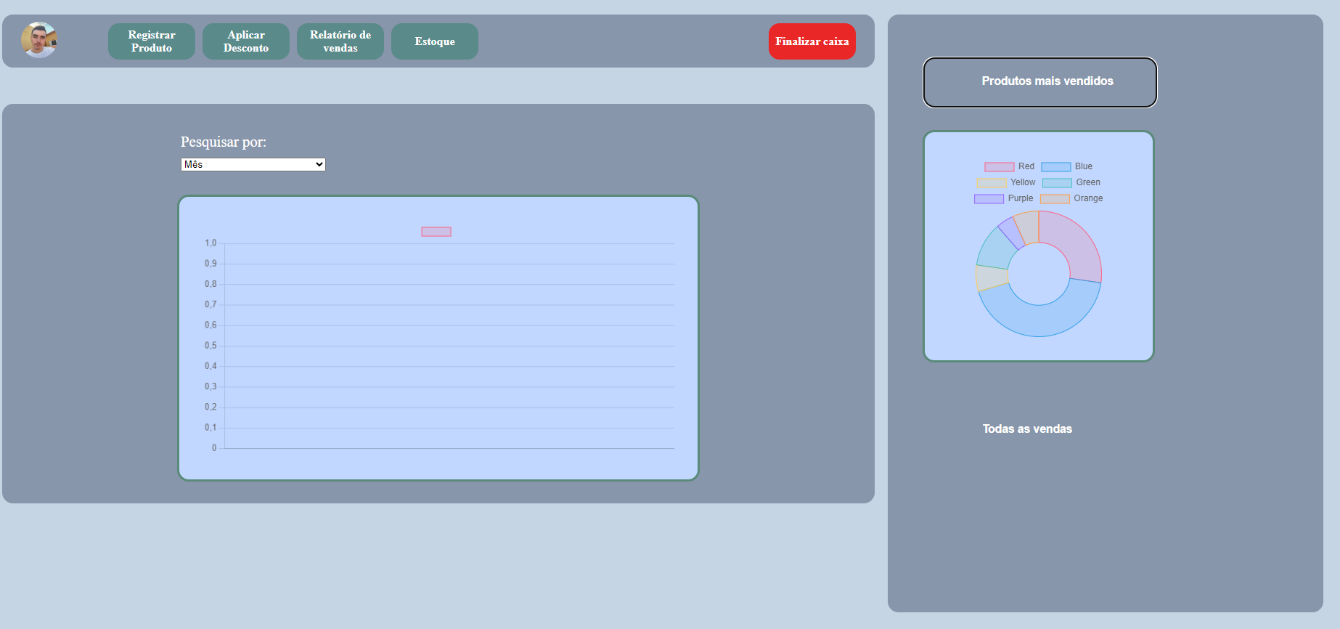












# CONCLUSÃO

# Escreva os resultados obtidos

Resultados obtidos

# Constatações

Constatações

# Sugestões de possíveis aperfeiçoamentos técnicos

Sugestões

# REFERÊNCIAS

Aqui vai o texto de referências (MORE)

# GLOSSÁRIO

Se houver necessidade

# ANEXOS

Se houver necessidade