Projeto Integrador 2º Semestre - DSM

**Disciplinas:**

Banco de Dados

Desenvolvimento Web II

Engenharia de Software II

**Professores:**

Bruno

Nilton

Orlando

**Grupo(7) / Nome da Empresa:**

**Sistema: Marmitaria Fit**

|  |  |
| --- | --- |
| Integrante | **Papel Principal** |
| Adinam Conte Cardozo | Desenvolvimento do Banco de Dados |
| Izabely do Nascimento Rodrigues de Campos | Documentação I |
| Rayanne Gabriela da Silva Sebastiao Nunes | Documentação II |
| Raphael Reis Rodrigues da Silva | Desenvolvimento beck-end (PHP) |
| Vitor Eduardo B. Marciano | Desenvolvimento front-end |

**Fatec Araras**

**2024**

FICHA DE CONTROLE - PROJETO INTERDISCIPLINAR

DISCIPLINA CHAVE: Engenharia de Software II - PI II

PROFESSOR: Bruno Henrique de Paula Ferreira

GRUPO: Grupo 7 SEMESTRE: 2/2024

TÍTULO DO PROJETO: Comidas Fit

DATA DA APRESENTAÇÃO: 03/12/2024

NOTA:

INTEGRANTES DO GRUPO: Grupo 7

|  |  |
| --- | --- |
| Nome | Nota Individual |
| Adinam Conte Cardozo |  |
| Izabely do Nascimento Rodrigues de Campos |  |
| Rayanne Gabriela da Silva Sebastiao Nunes |  |
| Raphael Reis Rodrigues da Silva |  |
| Vitor Eduardo B. Marciano |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Araras, 03 de dezembro de 2024

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Professor Bruno Henrique de Paula Ferreira

**Sumário**

[1. Apresentação da Empresa 4](#_Toc176466743)

[1.1 Missão 4](#_Toc176466744)

[1.4 Link Repositório 4](#_Toc176466745)

[2. Escopo do sistema 5](#_Toc176466746)

[2.1 Objetivos do projeto 5](#_Toc176466747)

[2.2 Técnica de levantamento de requisitos 5](#_Toc176466748)

[2.3 Requisitos funcionais 5](#_Toc176466749)

[2.4 Requisitos não funcionais 5](#_Toc176466750)

[2.5 Comparativo entre sistemas 6](#_Toc176466751)

[Descrever resumidamente e apresentar uma tabela de comparação do sistema desenvolvido versus 1 ou 2 outros sistemas de mercado. 6](#_Toc176466752)

[2.6 Cronograma 6](#_Toc176466753)

[3. Documentação do Sistema 7](#_Toc176466754)

[3.1 Metodologia de Desenvolvimento 7](#_Toc176466755)

[3.2 Diagramas UML 7](#_Toc176466756)

[Diagrama de caso de uso (1º) 7](#_Toc176466757)

[Diagrama de classes (2º) 7](#_Toc176466758)

[Diagrama de sequência (2º) 7](#_Toc176466759)

[3.3 Modelo Conceitual (1º) 8](#_Toc176466760)

[3.4 Modelo lógico (1º) 8](#_Toc176466761)

[3.5 Modelo Físico (2º) 8](#_Toc176466762)

[3.6 Recursos e ferramentas (1º e 2º) 8](#_Toc176466763)

[3.7 Etapas / Sprints realizados (1º e 2º) 8](#_Toc176466764)

[3.8 Interface do usuário 8](#_Toc176466765)

[4. Testes e Qualidade 9](#_Toc176466766)

[4.1 Estratégia de Testes: Descrever a estratégia de testes adotada 9](#_Toc176466767)

[4.2 Resultados dos Testes: Apresentar os resultados dos testes realizados 9](#_Toc176466768)

[4.3 Garantia da Qualidade: Descrever as práticas adotadas para garantia da qualidade 9](#_Toc176466769)

[4.4 Requisitos mínimos de hardware e software para o sistema 9](#_Toc176466770)

[4.5 Contrato para desenvolvimento de software 9](#_Toc176466771)

[5. Considerações Finais 9](#_Toc176466772)

[6 Referências 9](#_Toc176466773)

[Anexo I - Diário de bordo 9](#_Toc176466774)

[Anexo II – Cronograma efetivo 9](#_Toc176466775)

[Anexo III – Evidências 9](#_Toc176466776)

# Apresentação da Empresa

**ControlPlus – empresa de sistemas de controle de comércios pequenos.**

Ícone

Descrição gerada automaticamente

**Natural Leve – empresa que escolhemos para desenvolver o sistema**



## 1.1 Missão

Propor soluções tecnológicas inovadoras e acessíveis que simplifiquem a administração de produtos, inventário e plataformas online, promovendo a eficiência e o crescimento sustentável dos negócios de nossos clientes.

## 1.2 Visão

Ser reconhecida pela inovação, credibilidade e influência positiva na transformação digital das organizações, simplificando a administração de produtos, inventários e contribuição para a transformação digital das empresas.

## 1.3 Valores

Os princípios básicos que direcionam a conduta e as escolhas da empresa são a inovação em busca de novas ideias para revolucionar a administração de empresas, excelência na busca por soluções eficazes e de alta qualidade, simplicidade na criação de sistemas intuitivos e acessíveis, e o foco no cliente, com o objetivo de fornecer soluções intuitivas e acessíveis.

## 1.4 Link Repositório

<https://github.com/VitorMarciano001/PI_2_semestre.git>

# Escopo do sistema

O propósito do sistema de marmitaria é automatizar e aprimorar processos operacionais, garantindo uma gestão eficiente e integrada. Ao substituir processos manuais por funcionalidades automatizadas, o sistema busca reduzir erros, aumentar a produtividade e melhorar a experiência dos clientes.

## Objetivos do projeto

A meta do projeto é criar um sistema para a marmitaria, que possa aprimorar e integrar processos como o registro de pedidos, administração de estoque e geração de relatórios financeiros. O objetivo é trocar os processos manuais por uma plataforma digital eficaz, garantindo maior rapidez na operação e minimizando falhas. Adicionalmente, o sistema visa melhorar a satisfação dos clientes, oferecendo personalização de pedidos, diversas opções de pagamento e, no futuro, assistência para pedidos online. Assim, o objetivo do projeto é fomentar uma administração mais estruturada e estratégica, simplificando a administração da empresa.

## 2.2 Técnica de levantamento de requisitos

Para o levantamento de requisitos do sistema da marmitaria, foram utilizadas algumas técnicas, cada uma com o objetivo de capturar as necessidades específicas dos envolvidos e entender melhor os processos operacionais.

A entrevista foi uma das principais estratégias, realizada com o proprietário e os colaboradores. Para entender a rotina de produção, controle financeiro, administração de pedidos, realizamos algumas perguntas. A conversa com os colaboradores possibilitou a identificação de desafios específicos no controle de pedidos e na administração do estoque. Com a entrevista realizada, conseguimos entender como os pedidos eram elaborados e administrados, além de acompanhar o fluxo do inventário e as fases das entregas.

Foi realizado também um brainstorming com a equipe, reunindo o dono e os funcionários para discutirem ideias e priorizarem funcionalidades para o sistema. Essa abordagem colaborativa permitiu a criação de uma lista de melhorias com base nas necessidades mais importantes e nas sugestões de todos os envolvidos.

Por fim, foi realizada uma análise de documentos e processos existentes, como planilhas e anotações usadas para o controle manual de pedidos e estoque. A partir dessa análise, ficou evidente a necessidade de automatizar esses processos, integrando o controle de estoque e pedidos em um sistema único.

Com a aplicação dessas técnicas, foi possível mapear os processos da marmitaria, identificar os pontos de melhoria e definir os requisitos essenciais para o sistema. O levantamento garantiu uma visão completa das necessidades operacionais e permitiu que o desenvolvimento do sistema fosse orientado por informações precisas e relevantes, promovendo maior eficiência e satisfação dos usuários.

## Requisitos funcionais

Requisitos funcionais, são declarações dos serviços que o sistema deve fornecer, do modo como o sistema deve reagir a determinadas entradas e de como deve se comportar em determinadas situações. (Sommerville)

|  |  |
| --- | --- |
| **Requisito** | Cadastro |
| **Identificador** | REQF01 |
| **Descrição** | O sistema deve permitir que somente o usuário administrador faça cadastro de novos usuários ao sistema, pedindo as seguintes informações, nome, e-mail para cadastro, senha, nível de acesso e devera guardar a data de cadastro |
| **Prioridade** | Essencial |
| **Dependência** | Não ha dependências Relacionadas |
| **Restrições** | Não há restrições |
| **Ator(es)** | Administrador |

|  |  |
| --- | --- |
| **Requisito** | Login |
| **Identificador** | REQF02 |
| **Descrição** | O sistema deve permitir que o usuário autentique-se utilizando suas credenciais (usuário e senha) para acessar as funcionalidades disponíveis após o login. |
| **Prioridade** | Essencial |
| **Dependência** | REQF01 |
| **Restrições** | Não há restrições |
| **Ator(es)** | Usuário/Administrador |

|  |  |
| --- | --- |
| **Requisito** | Cadastro de ingredientes |
| **Identificador** | REQF03 |
| **Descrição** | O sistema deve permitir o cadastro de novos ingredientes no estoque   * Devem ter os seguintes campos nome, categoria, fornecedor, quantidade, preço compra, data validade. * O sistema deve permitir a edição e exclusão de ingredientes já cadastrados. |
| **Prioridade** | Essencial |
| **Dependência** | Não há dependência relacionada |
| **Restrições** | Não há restrição relacionada |
| **Ator(es)** | Usuário/Administrador |

|  |  |
| --- | --- |
| **Requisito** | Controle de Entrada de Estoque |
| **Identificador** | REQF04 |
| **Descrição** | O sistema deve registrar a entrada de novos ingredientes no estoque.   * O usuário pode informar o ingrediente, quantidade adicionada, fornecedor e data da entrada. * O sistema deve atualizar automaticamente a quantidade disponível no estoque após o registro da entrada. |
| **Prioridade** | Essencial |
| **Dependência** | Não há dependência |
| **Restrições** | Não há restrição |
| **Ator(es)** | Usuário/Administrador |

|  |  |
| --- | --- |
| **Requisito** | Controle de Saída de Estoque |
| **Identificador** | REQF04 |
| **Descrição** | O sistema deve registrar a saída de ingredientes do estoque para produção de marmitas.   * Terá um botão para escolher qual ingrediente será excluído, lá ele onde mostrará a o registro do item podendo alterá-lo * O sistema deve deduzir automaticamente a quantidade retirada do estoque com baste na comanda. |
| **Prioridade** | Essencial |
| **Dependência** | REQF03 |
| **Restrições** | O produto deve ter sido cadastrado no sistema |
| **Ator(es)** | Usuário/Administrador |

|  |  |
| --- | --- |
| **Requisito** | Alerta de Baixo Estoque |
| **Identificador** | REQF05 |
| **Descrição** | O sistema deve notificar o usuário quando o nível de estoque de um ingrediente atingir um limite mínimo.   * O sistema deve permitir que o usuário defina um nível mínimo para cada ingrediente. * Quando o estoque atingir esse nível, uma notificação será gerada na interface e/ou enviada por e-mail. |
| **Prioridade** | Média |
| **Dependência** | Não há dependência |
| **Restrições** | Não há Restrição |
| **Ator(es)** | Administrador |

|  |  |
| --- | --- |
| **Requisito** | Relatórios de Estoque |
| **Identificador** | REQF06 |
| **Descrição** | O sistema deve gerar relatórios de movimentação de estoque.   * O usuário pode gerar relatórios diários, semanais ou mensais de entradas e saídas de estoque. * Relatórios personalizados podem ser gerados com base em categorias de ingredientes, fornecedores ou períodos específicos. |
| **Prioridade** | Essencial |
| **Dependência** | Não há dependência |
| **Restrições** | Não há Restrição |
| **Ator(es)** | Usuário/Administrador |

|  |  |
| --- | --- |
| **Requisito** | Comanda de pedidos |
| **Identificador** | REQF06 |
| **Descrição** | O sistema terá uma funcionalidade onde o usuário poderá adicionar novos pedidos, funcionando como uma comanda digital para as marmitas. Para cada novo pedido, os seguintes campos serão **obrigatórios**:   * Nome do Cliente: Nome do cliente que faz o pedido. * Nome da Marmita: O nome da marmita que o cliente deseja. * Data de Entrega: A data prevista para a entrega do pedido.   **Não obrigatório**   * Responsável |
| **Prioridade** | Média |
| **Dependência** | Não há dependência |
| **Restrições** | Não há Restrição |
| **Ator(es)** | Usuário/Administrador |

## 2.4 Requisitos não funcionais

Requisitos não funcionais são restrições sobre os serviços ou funções oferecidas pelo sistema. Eles incluem restrições de tempo, restrições sobre o processo de desenvolvimento e restrições impostas por padrões. Os requisitos não funcionais se aplicam, frequentemente, ao sistema como um todo, em vez de às características individuais ou aos serviços. (Sommerville)

|  |  |
| --- | --- |
| **Requisito** | Segurança dos Dados |
| **Identificador** | RNF01 |
| **Descrição** | * O sistema deve criptografar todos os dados sensíveis (como dados de fornecedores e quantidades de estoque) antes de armazená-los no banco de dados. |
| **Prioridade** | Essencial |
| **Dependência** | Não há dependência |
| **Restrições** | Devem ser seguidas as diretrizes de segurança da LGPD (Lei Geral de Proteção de Dados). |
| **Ator(es)** | Usuário/Administrador |

|  |  |
| --- | --- |
| **Requisito** | **Usabilidade** |
| **Identificador** | RNF02 |
| **Descrição** | * O sistema deve ser intuitivo, permitindo que um usuário novo consiga realizar as principais funções (cadastro, entrada e saída de estoque) sem necessidade de treinamento formal em até 30 minutos de uso. |
| **Prioridade** | Média |
| **Dependência** | Não há dependência |
| **Restrições** | Não há restrições |
| **Ator(es)** | Usuário/Administrador |

## 2.5 Comparativo entre sistemas

Fazendo um comparativo com os sitemas Bling e Awise, sistemas de gestão e controle de estoque, percebemos as seguintes características:

1. **Nosso sistema:** Fornece controle básico de estoque, sem integrações ou funcionalidades financeiras, ideal para pequenas operações com necessidades limitadas.
2. **Bling ERP:** Uma solução robusta que integra gestão avançada de estoque, controle financeiro e múltiplas integrações com plataformas de e-commerce e marketplaces. Recomendado para empresas com operações complexas e múltiplos canais de venda.
3. **Awise:** Sistema intermediário que foca em integrações com vendas online, combinando funcionalidades básicas de estoque e finanças. Adequado para pequenas e médias empresas que buscam otimizar a gestão de vendas.

Em suma, o Bling é mais adequado para empresas que demandam uma gestão abrangente, enquanto o Awise serve melhor negócios menores com foco em comércio eletrônico.

Decidimos desenvolver um sistema simples para o nosso pequeno negócio de marmitaria, focando nas funcionalidades essenciais que atendem as principais necessidades. Essa abordagem resultou em um sistema que se assemelha ao Awise em funcionalidades, que é projetado especificamente para pequenas empresas, oferecendo a praticidade e eficiência que buscamos.

## Tabela 1 – Comparativo das funcionalidades da aplicação

|  |  |  |  |
| --- | --- | --- | --- |
| **Funcionalidades** | **Bling** | **Awise** | **Meu Sistema** |
| **Gerenciar estoque** | X | X | X |
| **Cadastro de produtos** | X | X | X |
| **Gestão de clientes e fornecedores** | X | X | - |
| **Entrada de estoque** | X | X | X |
| **Saída de estoque** | X | X | X |
| **Controle de lotes** | X | - | - |
| **Controle de validade** | X | - |  |
| **Alerta de baixo estoque** | X | X | X |
| **Relatório de estoque** | X | X | X |
| **Integração com outros sistemas** | X | X | - |

## 2.6 Cronograma

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Tarefas** | **22/08** | **29/08 a 05/09** | **12/09 a 26/09** | **01/10 a**  **22/10** | **29/10 a 12/11** | **13/11 a 22/11** | **03/12** |
| Definição Grupos | X |  |  |  |  |  |  |
| Criação Empresa |  | X |  |  |  |  |  |
| Escopo Sistema |  |  | X |  |  |  |  |
| Requisitos |  |  | X | X | X |  |  |
| Diagramas |  |  | X | X | X |  |  |
| Protótipo |  |  |  |  | X | X |  |
| Documentação |  | X | X | X | X | X |  |
| Entrega |  |  |  |  |  | X |  |
| Apresentação |  |  |  |  |  |  | X |

# Documentação do Sistema

Neste capítulo, são apresentados os documentos técnicos que descrevem os aspectos fundamentais do sistema desenvolvido, fornecendo uma base sólida para compreensão e manutenção futura. A documentação é uma parte essencial do processo de desenvolvimento de software, pois oferece um registro detalhado das decisões tomadas e das características do sistema.

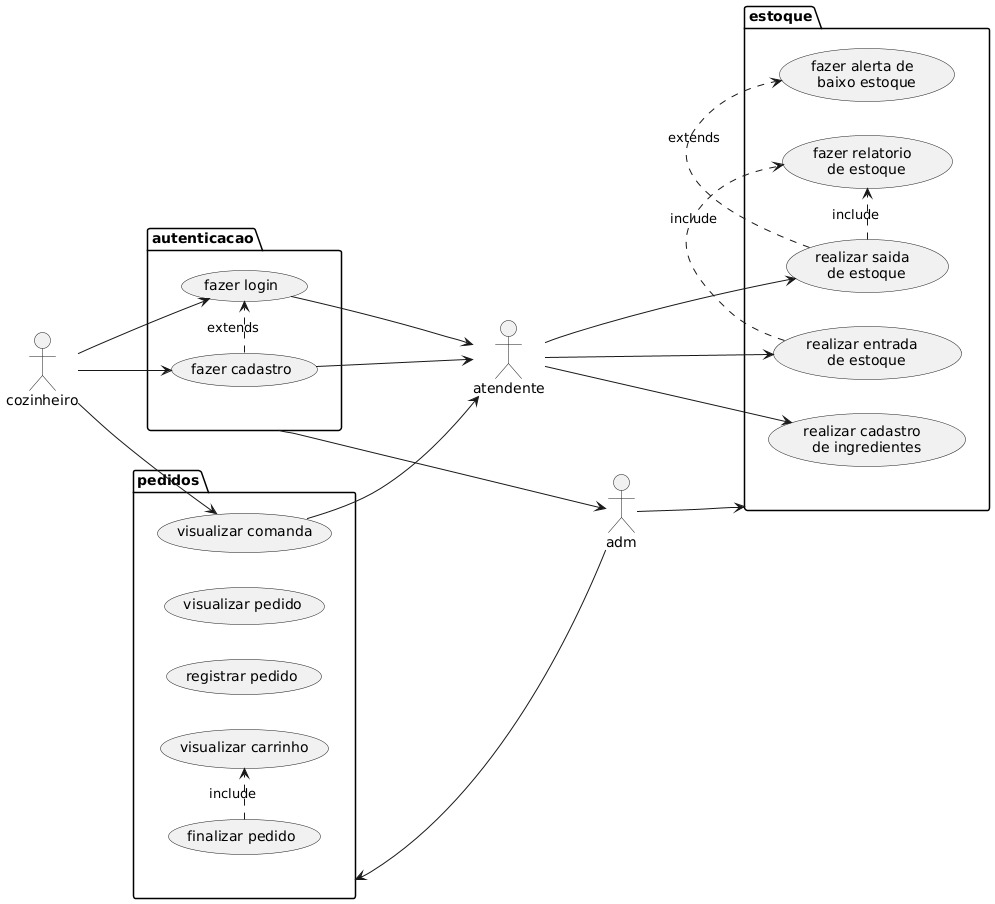
## Metodologia de Desenvolvimento

O Scrum é uma metodologia de desenvolvimento ágil fundamentada em sprints, os períodos de produção de um projeto, assegurando uma revisão e aprimoramento contínuos para assegurar que o resultado final seja sempre o mais excelente.   
A colaboração é essencial na implementação desta estratégia.  
Um dos frameworks mais utilizados, a razão é bastante clara: sua simplicidade. Através de ciclos de produção mais curtos, os gestores conseguem monitorar a progressão de um projeto de maneira gradual. Portanto, com revisões frequentes, o resultado final do trabalho tende a ser consideravelmente superior.

Utilizamos dessa metodologia pois achamos que seria a melhor opção para nós, por ela ser simples para ser aplicada na execução do nosso projeto.

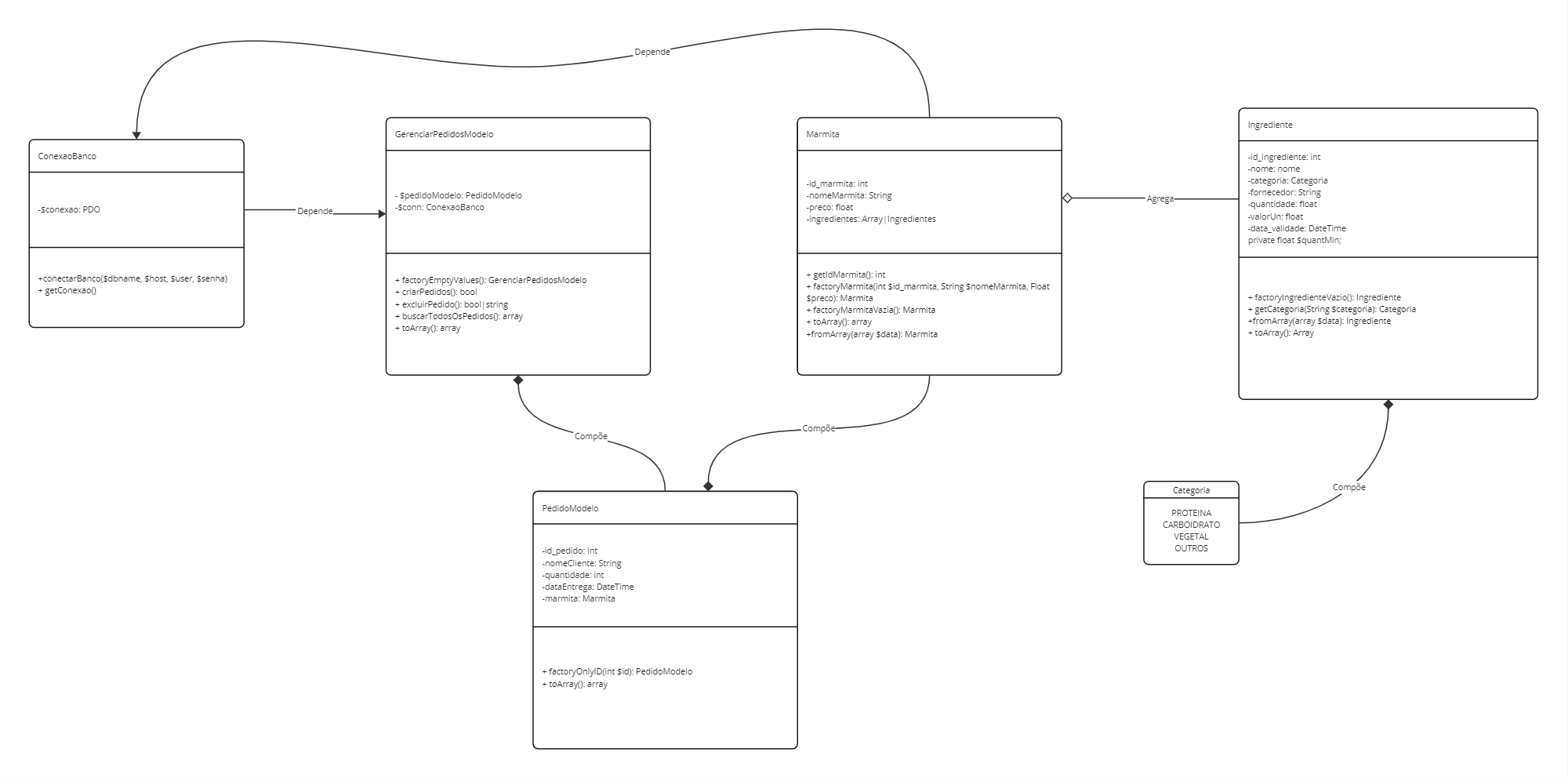
## 3.2 Diagramas UML

## Diagrama de caso de uso (1º)

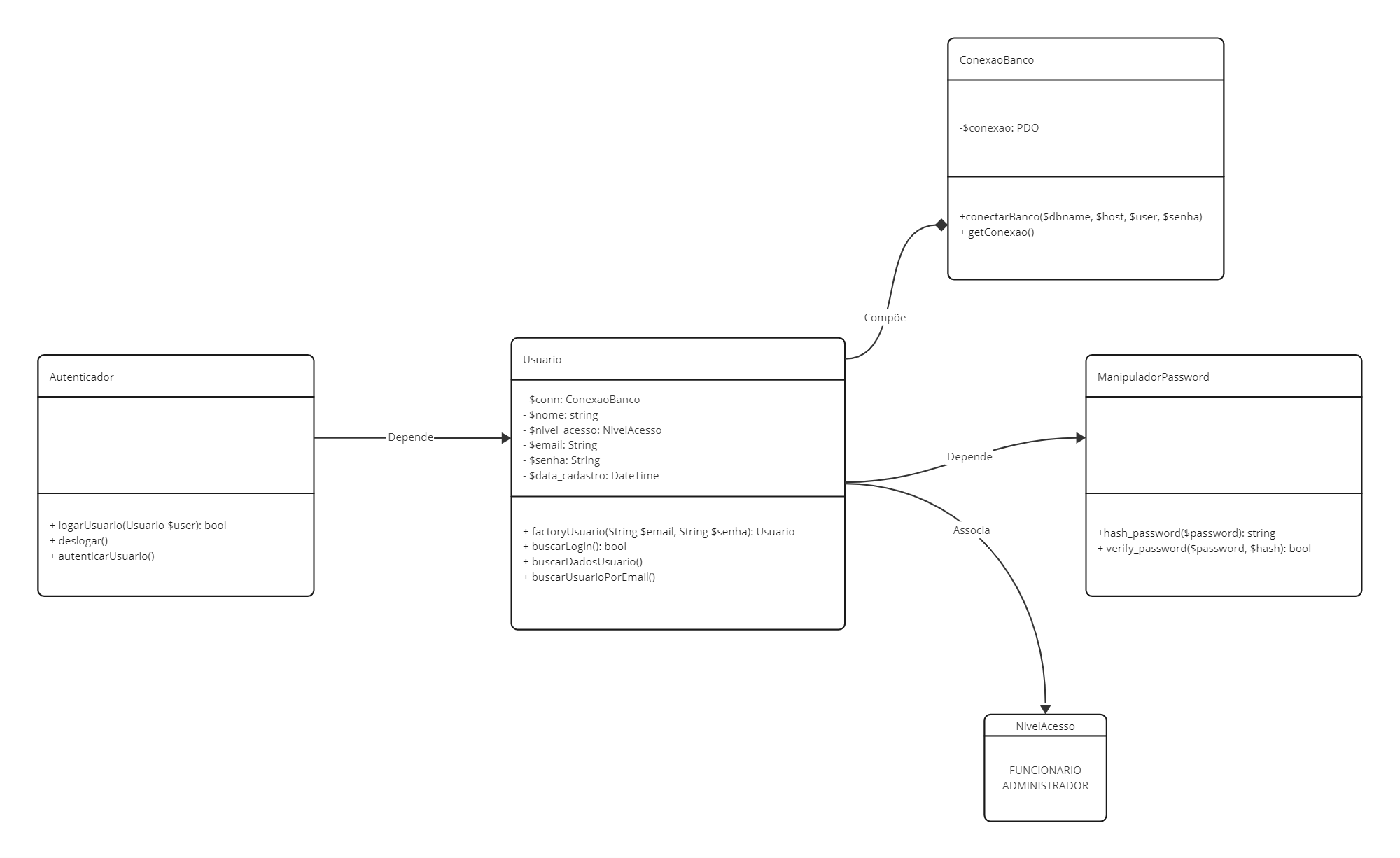


## Diagrama de classes (2º)

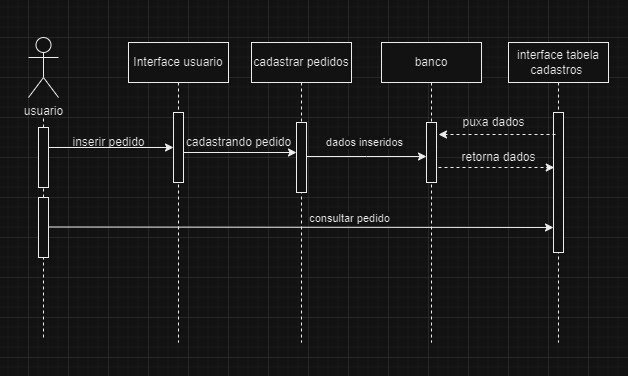
**Diagrama de classe Gerenciamento de pedidos**



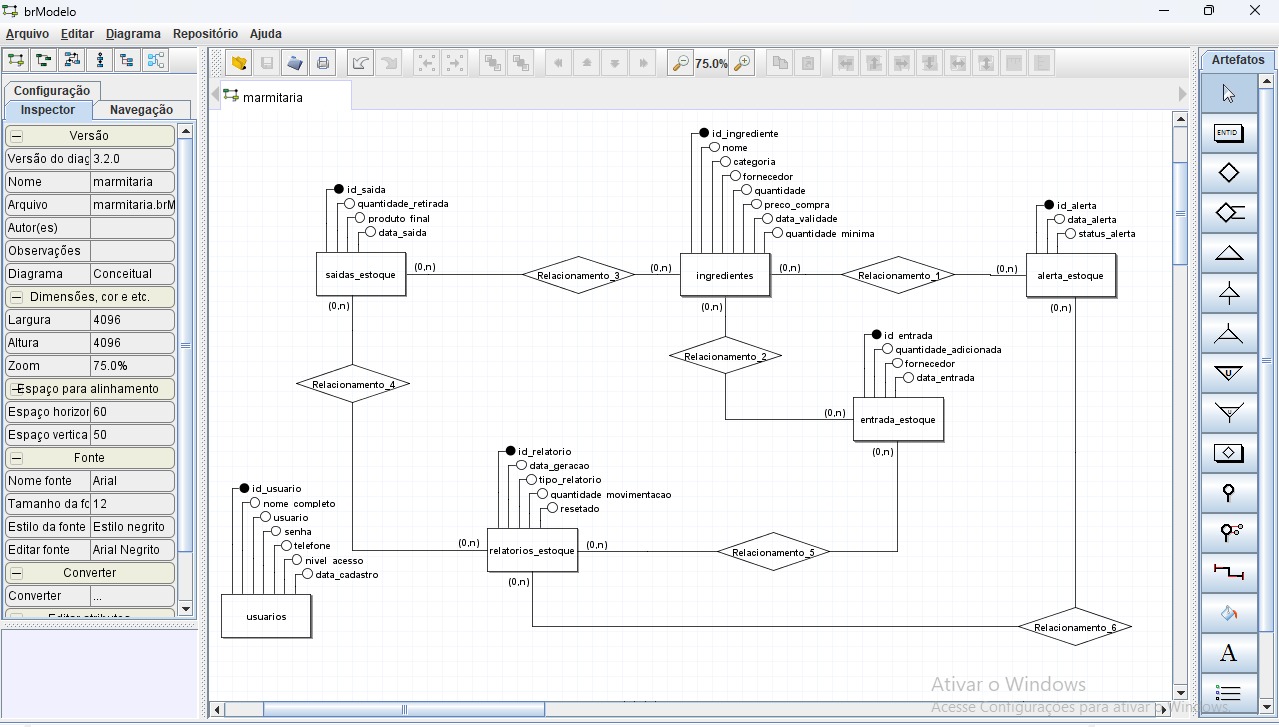
**Diagrama de classe login**



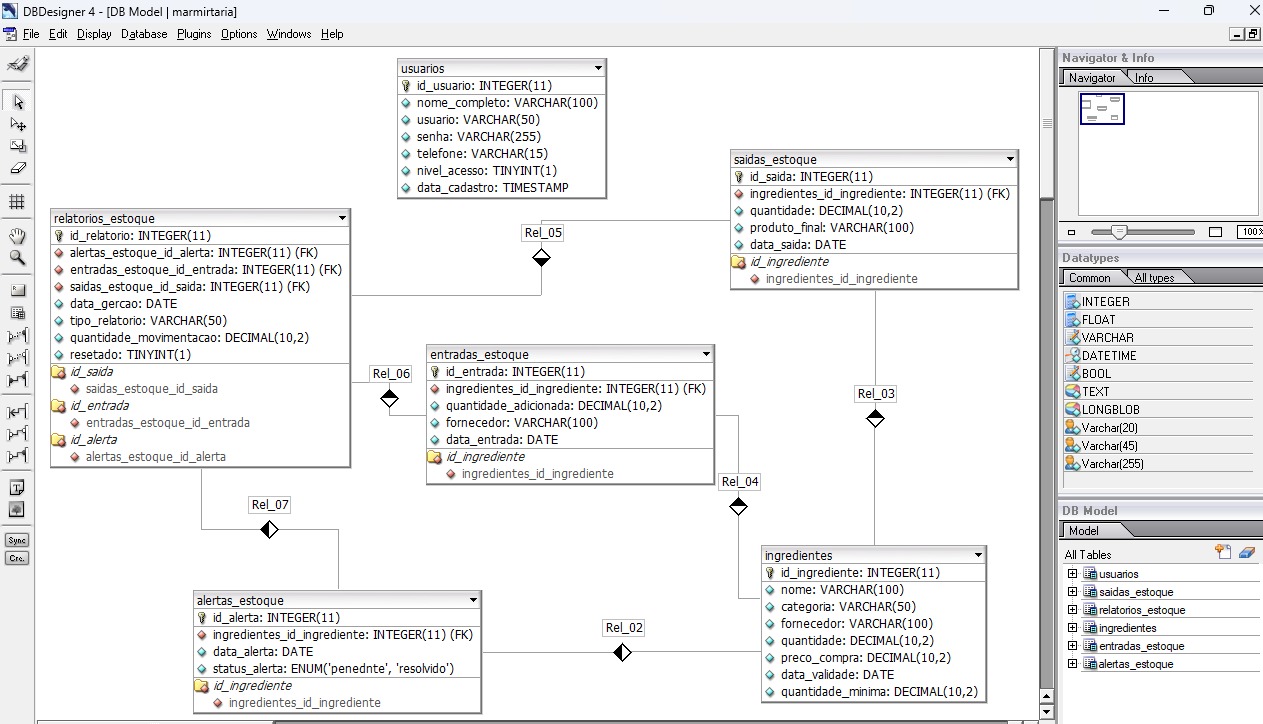
## Diagrama de sequência (2º)



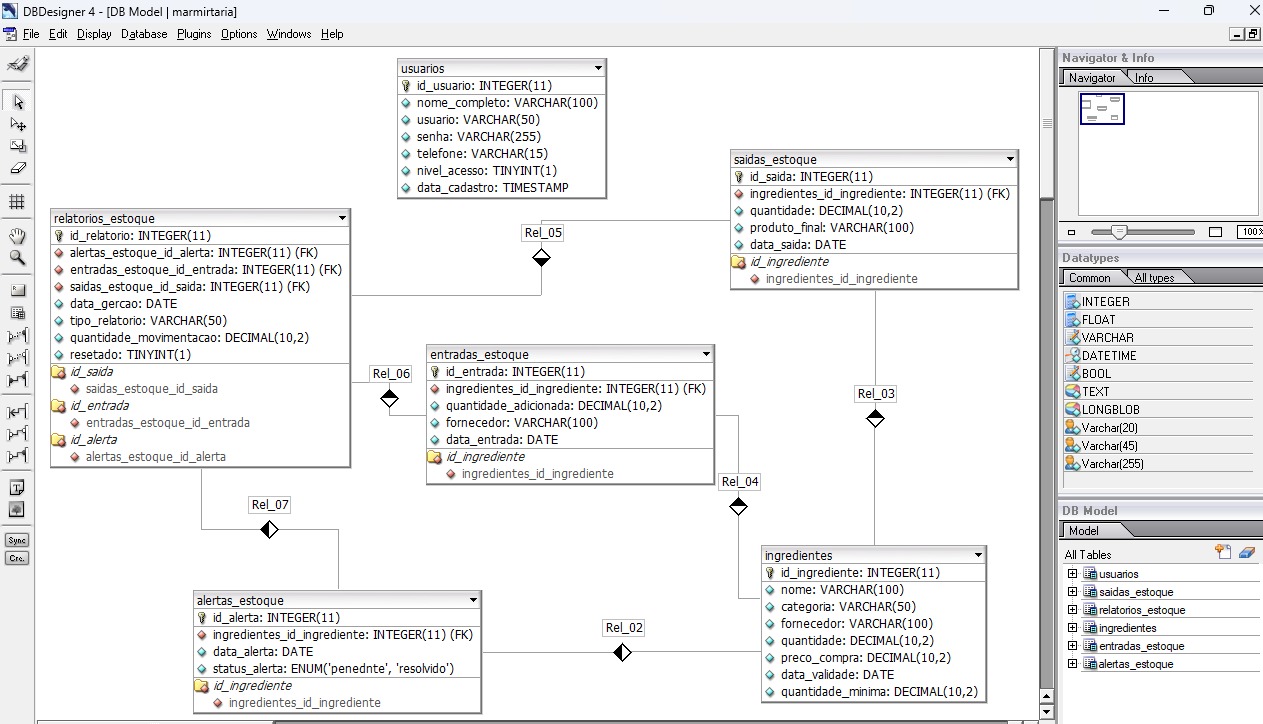
## Modelo Conceitual (1º)



## Modelo lógico (1º)



## Modelo Físico (2º)



## Recursos e ferramentas (1º e 2º)

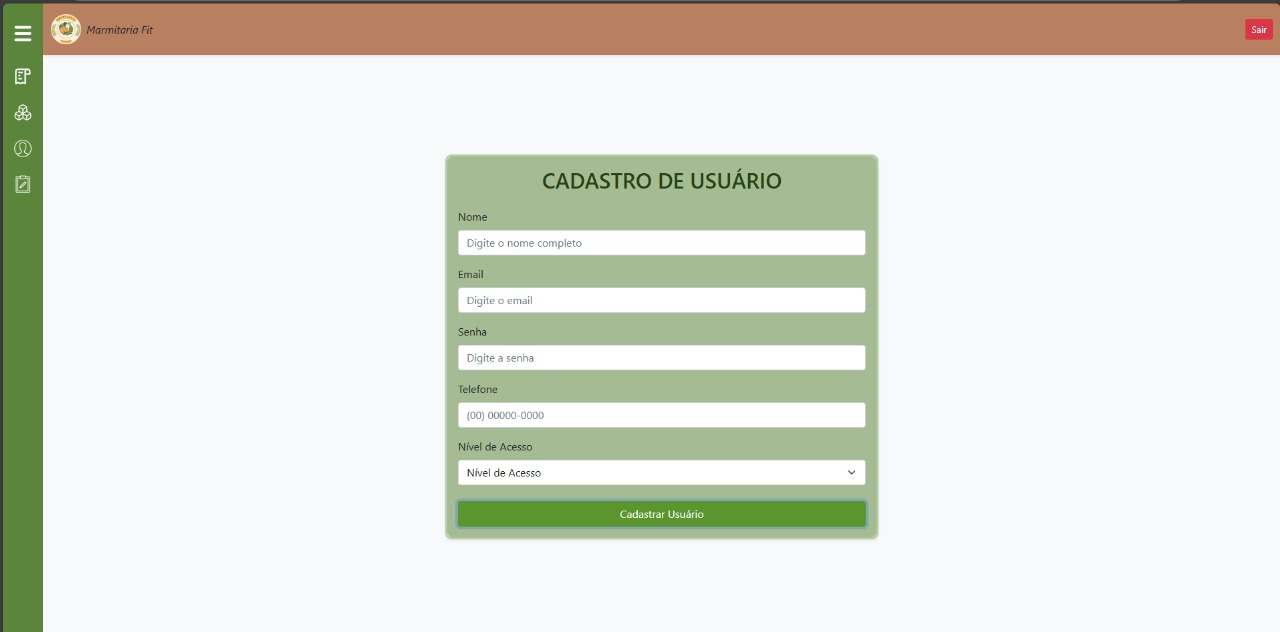
## 3.7 Etapas / Sprints realizados (1º e 2º)

## 3.8 Interface do usuário

**Tela de login**



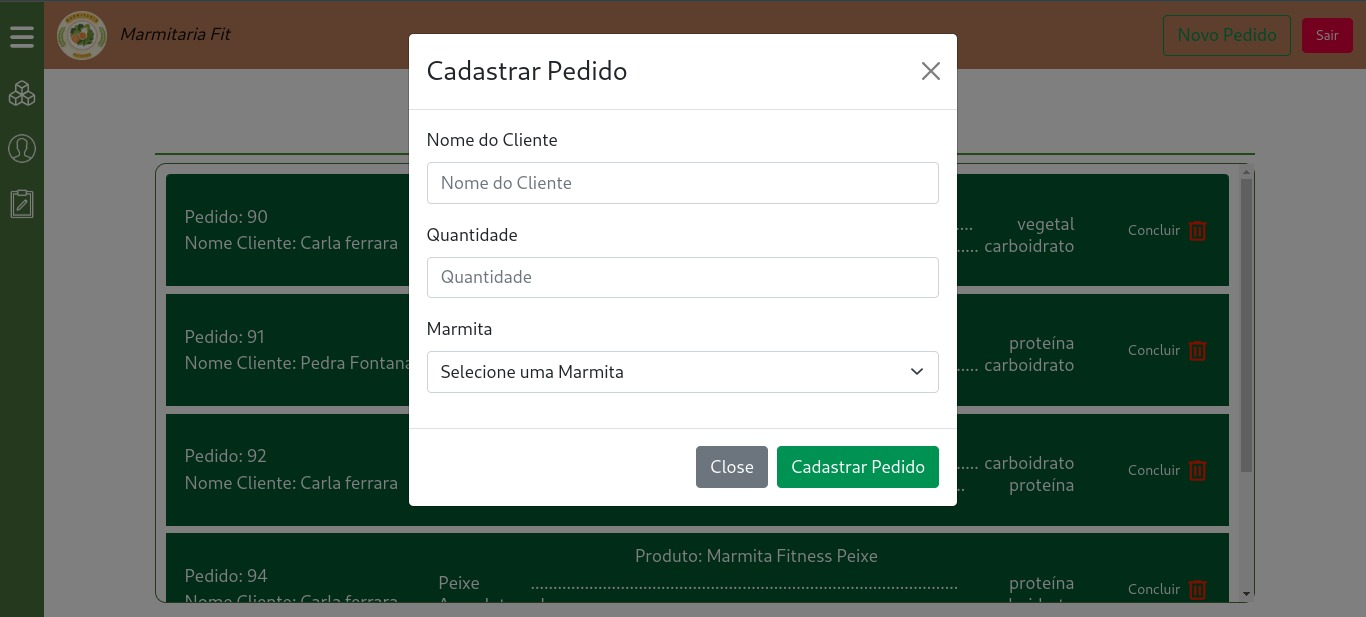
**Tela de cadastro de usuario**



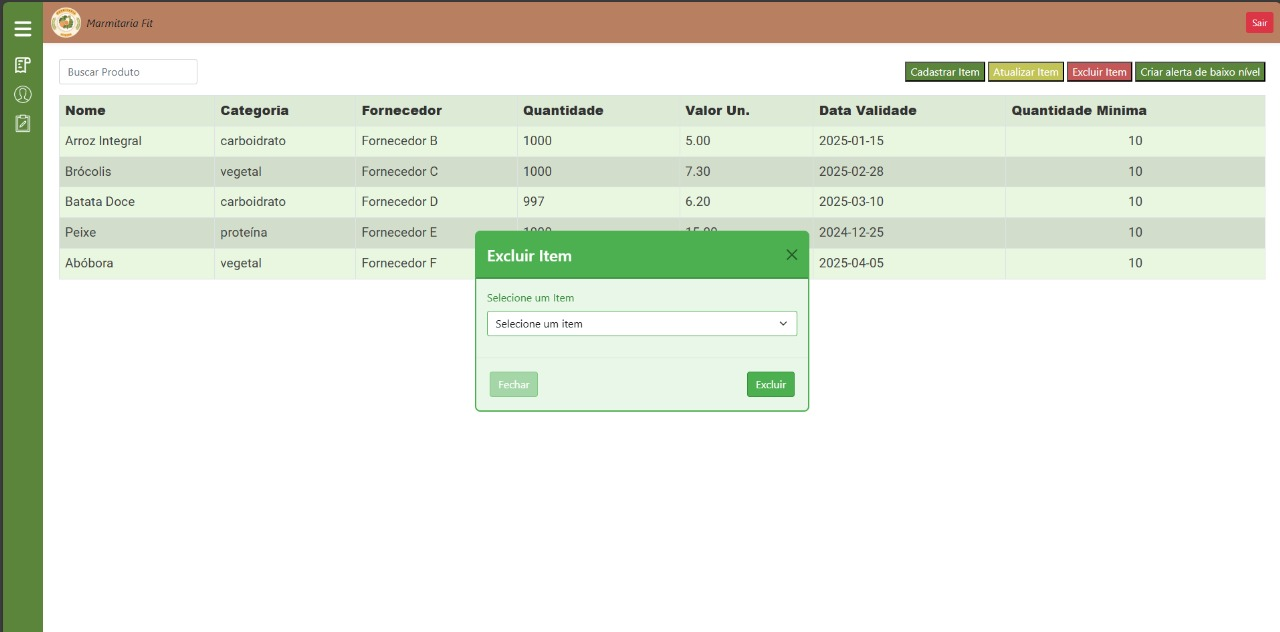
**Tela de cadastro de marmita**



**Tela de cadastrar Pedido**



**Tela de excluir item**



# Testes e Qualidade

## 4.1 Estratégia de Testes: Descrever a estratégia de testes adotada

Testes foram realizados durante a execução dos codigos. Durante a produção de todo sistema de gerenciamento, os responsáveis pelo codigo e interfaces (Raphael e Vitor) foram realizando testes simples, como registrar pedidos, usuarios..., por conta propria, para ter certeza se seria funcional.

## 4.2 Resultados dos Testes: Apresentar os resultados dos testes realizados

## 4.3 Garantia da Qualidade: Descrever as práticas adotadas para garantia da qualidade

Para assegurar a qualidade do sistema, além dos testes, foi necessário estabelecer requisitos e trabalhar de forma colaborativa com os envolvidos. A implementação de metodologias, apesar de terem sido implementadas poucas, possibilitou o acompanhamento do avanço. As revisões de código contribuíram para detectar erros e assegurar a conformidade. A documentação garantiu o rastreamento e a compreensão do sistema.

## 4.4 Requisitos mínimos de hardware e software para o sistema

## 4.5 Contrato para desenvolvimento de software

# Considerações Finais

O desenvolvimento do sistema de gestão para a Natural leve (Na qual deixo claro que de primeira não aceitou conversar conosco, porém com um pouco de insistencia responderam às perguntas necessárias) permitiu ao grupo aplicar na prática os conhecimentos adquiridos ao longo do semestre, envolvendo áreas como banco de dados, desenvolvimento web e engenharia de software. Trabalhamos de forma colaborativa para superar desafios e criar uma solução eficiente para a gestão de pedidos e estoque de uma marmitaria, atendendo às necessidades identificadas durante o levantamento de requisitos.

A experiência reforçou a importância de planejamento, organização e testes contínuos para garantir a qualidade do sistema. Além de desenvolver habilidades técnicas, aprendemos sobre o trabalho em equipe e o uso de metodologias ágeis para alcançar os objetivos do projeto. O resultado foi um sistema funcional que atende aos requisitos propostos, contribuindo para o aprendizado e crescimento de todos os integrantes.

# 6 Referências

# Anexo I - Diário de bordo

**Físico:** Feito pelo grupo.

**Digital:** individual / Links

# Anexo II – Cronograma efetivo

Não houve cronograma.

# Anexo III – Evidências

Não houve prints pois todas as conversas, desde a conversa com a cliente até a conversa dos integrantes do grupo, foram todas feitas pessoalmente.