



UNIVERSIDADE FEDERAL DO MARANHÃO
BACHARELADO INTERDISCIPLINAR EM CIÊNCIA E TECNOLOGIA

ANA IARA LOAYZA COSTA
BRUNO RAFAEL CARVALHO COELHO
JOÃO LUCAS SILVA DA SILVA
JOSÉ VICTOR BRITO COSTA
MILENA FREIRE BRITTO NEVES

DOCUMENTAÇÃO - SISTEMA DE MERCADO

SÃO LUÍS - MA
DEZEMBRO/2023

ANA IARA LOAYZA COSTA (2021017729)
BRUNO RAFAEL CARVALHO COELHO (2022024791)
JOÃO LUCAS SILVA DA SILVA (2021017890)
JOSÉ VICTOR BRITO COSTA (2021017925)
MILENA FREIRE BRITTO NEVES (2021018029)

DOCUMENTAÇÃO - SISTEMA DE MERCADO

Documento apresentado como requisito parcial de avaliação da disciplina Laboratório de Projeto e Desenvolvimento de Software - Turma 02, no curso Bacharelado Interdisciplinar em Ciência e Tecnologia da Universidade Federal do Maranhão.

Orientador: Prof. Dr. Thales Levi Azevedo Valente.

SÃO LUÍS - MA
DEZEMBRO/2023

RESUMO

Este documento tem como propósito apresentar o desenvolvimento do sistema de mercado. Seu principal propósito prático é facultar o controle eficiente de estoque, proporcionando um acesso controlado e facilitado para os colaboradores de uma mercearia, resultando na otimização da gestão diária do estabelecimento. Além disso, busca-se a automação aprimorada do gerenciamento de estoque, garantindo um controle meticuloso sobre os produtos disponíveis, com o intuito de aprimorar a eficiência operacional e viabilizar um acompanhamento detalhado e eficaz dos produtos em questão.

Palavras-chaves: controle, gerenciamento, estoque, sistema de mercado

SUMÁRIO

1. INTRODUÇÃO.....	5
2. OBJETIVOS.....	6
3. ELICITAÇÃO DE REQUISITOS.....	7
3.1 Requisitos Funcionais.....	7
3.2 Requisitos Não Funcionais.....	8
4. DIAGRAMA DE CASO DE USO.....	11
4.1 Diagramação.....	11
4.2 Descrição dos Casos de Uso.....	13
4.2.1 Realizar Login.....	13
4.2.2 Controlar Estoque.....	14
4.2.3 Visualizar Dashboard.....	15
4.2.4 Registrar Vendas.....	16
4.2.5 Editar Usuário Pessoal.....	17
4.2.6 Editar Usuários.....	18
5. DIAGRAMA DE CLASSE.....	19
6. DIAGRAMA DE SEQUÊNCIA.....	21
7. DIAGRAMA DE ATIVIDADE.....	24
7.1 Realizar Login.....	24
7.2 Registrar Vendas.....	25
7.3 Controlar Estoque.....	26
7.4 Visualizar Dashboard.....	27
7.5 Editar Usuário Pessoal.....	28
7.6 Editar Usuários.....	29
8. TECNOLOGIAS USADAS.....	30
9. ESCOPO DA PROTOTIPAÇÃO.....	32
9.1 Login.....	32
9.2 Dashboard.....	33
9.2 Estoque.....	34
9.2 Usuário.....	37
10. CONCLUSÃO.....	40
REFERÊNCIAS BIBLIOGRÁFICAS.....	41

1. INTRODUÇÃO

Este documento tem como objetivo principal capturar e definir os requisitos necessários para o desenvolvimento do sistema de gestão destinado à mercearia Sunny. Os clientes, um casal de idosos proprietários do mercado, expressaram a necessidade urgente de um sistema que possa otimizar a gestão diária de vendas e estoque, fornecendo um controle eficaz para eles e seus funcionários.

Além disso, o documento apresentará diagramas de caso de uso, de classe, de sequência e de atividade, os quais desempenham um papel crucial na modelagem visual dos requisitos. Esses diagramas proporcionam uma compreensão clara das interações e estrutura do sistema, sendo fundamentais para facilitar a comunicação entre os stakeholders e a equipe de desenvolvimento, garantindo uma implementação eficaz e alinhada com as necessidades dos clientes.

A aplicação proposta será um sistema abrangente de gestão que visa automatizar as operações diárias da mercearia Sunny. O software permitirá o gerenciamento de vendas, controle de estoque, a visualização de dashboards que apresenta as principais tendências do mercadinho e possui acesso restrito para os funcionários. Com uma interface intuitiva, o sistema será projetado para simplificar as tarefas diárias, proporcionando maior eficiência operacional.

2. OBJETIVOS

O projeto visa implementar um sistema de gestão abrangente para a mercearia Sunny, atendendo à demanda urgente dos proprietários, um casal de idosos, por uma solução eficiente para otimizar as operações diárias. Com o aumento da clientela, a necessidade de um controle preciso do estoque e uma gestão simplificada das vendas tornou-se crucial. O software proposto será uma aplicação intuitiva que automatiza o gerenciamento de vendas e estoque, garantindo maior eficiência operacional.

Principais objetivos do projeto:

1. Automatização e Controle:

Implementar funcionalidades que automatizem o registro de vendas e mantenham um controle eficaz do estoque, permitindo uma gestão mais eficiente.

2. Facilitação dos Processos:

Proporcionar uma interface intuitiva para simplificar o registro de vendas, tornando o processo acessível tanto para clientes quanto para funcionários.

3. Informações Estratégicas:

Fornecer relatórios detalhados sobre as vendas para auxiliar os proprietários na tomada de decisões estratégicas para o crescimento do negócio.

4. Eficiência Operacional:

Otimizar as operações diárias, reduzindo erros e economizando tempo, para que os proprietários possam focar em atividades mais estratégicas.

5. Segurança e Controle de Acesso:

Garantir a segurança das informações e implementar um controle de acesso para funcionários, assegurando a confidencialidade dos dados.

3. ELICITAÇÃO DE REQUISITOS

Os requisitos funcionais e não funcionais levantados durante a entrevista com os clientes e posteriormente consolidados na etapa de levantamento de requisitos estão elencados a seguir.

3.1 Requisitos Funcionais

1. Sistema de Login e Cadastro de Funcionários:
 - Os funcionários e o administrador do mercadinho devem ser capazes de fazer login no sistema;
 - Deve haver a opção de registro de novos funcionários no sistema;
 - Prioridade: Essencial.
2. Visualizar Dashboard:
 - Exibição de um histórico de produtos mais vendidos, permitindo que os funcionários e o administrador identifiquem os produtos populares;
 - Gráfico que mostra estatísticas de vendas e tendências ao longo do tempo.
 - Prioridade: Essencial.
3. Adicionar Produto:
 - Os funcionários e o administrador devem ter a capacidade de adicionar novos produtos ao sistema, inserindo informações como nome, categoria, preço e quantidade disponível.
 - O sistema deve validar a entrada de dados para garantir a integridade das informações.
 - Prioridade: Essencial.
4. Editar Produto:
 - Funcionalidade que permite aos funcionários e ao administrador editar informações de produtos existentes, como preço, quantidade em estoque, categoria, e outros detalhes relevantes.
 - Alterações feitas devem ser refletidas em tempo real no sistema, garantindo consistência nos dados.
 - Prioridade: Essencial.

5. Remover Produto:

- Funcionalidade que permite a remoção de produtos do sistema quando necessário.
- Deve exigir confirmação para evitar exclusões acidentais e manter um histórico de alterações.
- Prioridade: Essencial.

6. Visualizar Produto:

- Os funcionários e o administrador devem poder visualizar detalhes completos de um produto, incluindo nome, categoria, preço, quantidade em estoque e outras informações relevantes.
- A busca e visualização eficiente são essenciais para facilitar a identificação rápida de produtos.
- Prioridade: Essencial.

7. Realizar Venda:

- Capacidade de registrar vendas, incluindo informações sobre produtos vendidos, quantidade, preços e clientes (se aplicável);
- Prioridade: Desejável.

8. Relatório de Vendas:

- Emissão de notas fiscais para cada venda;
- Histórico de vendas para consulta futura.
- Prioridade: Desejável.

Desse modo, os requisitos funcionais são cruciais para garantir a eficácia do sistema de gestão na mercearia Sunny. Eles abordam desde o cadastro de funcionários até o relatório final de vendas, garantindo que a equipe tenha as ferramentas necessárias para manter um controle preciso do estoque e das vendas, facilitando as operações diárias.

3.2 Requisitos Não Funcionais

1. Desempenho:

- O sistema deve ser capaz de lidar eficientemente com um volume significativo de transações, garantindo tempos de resposta rápidos durante a busca de informações de produtos, registro de vendas e outras operações críticas;

- A resposta do sistema não deve exceder um determinado limite, mesmo em momentos de pico de atividade;
- Prioridade: Essencial.

2. Segurança:

- Proteção dos dados de login e das informações das vendas para garantir a confidencialidade e a integridade dos dados;
- Recursos de autenticação segura, como senhas criptografadas;
- Prioridade: Essencial.

3. Disponibilidade:

- O sistema deve estar disponível para acesso praticamente ininterrupto, assegurando que os funcionários possam utilizar as funcionalidades essenciais sempre que necessário;
- Deve haver um plano de contingência para lidar com situações de falha e garantir uma rápida recuperação;
- Prioridade: Essencial.

4. Responsivo:

- O sistema deve ser responsivo em diferentes dispositivos, como computadores, tablets e smartphones, garantindo uma experiência de usuário consistente;
- As interfaces devem se adaptar dinamicamente ao tamanho da tela para facilitar a utilização em diversos contextos;
- Prioridade: Desejável.

5. Integração:

- Capacidade de integração com outros sistemas, como sistemas de pagamento, para facilitar transações suaves e proporcionar uma experiência completa aos clientes;
- Deve suportar padrões de integração relevantes ao contexto do mercadinho Sunny;
- Prioridade: Desejável.

6. Manutenção:

- Deve ser possível realizar atualizações e manutenções regulares no software sem causar interrupções significativas nas operações do mercado.
- Prioridade: Essencial.

7. Criptografia:

- As senhas salvas dos usuários devem ser criptografadas antes de serem armazenadas no banco de dados. Onde será necessário a validação sem a consulta da senha.
- Prioridade: Essencial.

Dessa maneira, os requisitos não funcionais são fundamentais para garantir que o sistema atenda não apenas às necessidades operacionais imediatas da mercearia Sunny, mas também aos requisitos de desempenho, segurança, integração e manutenção ao longo do tempo.

4. DIAGRAMA DE CASO DE USO

4.1 Diagramação

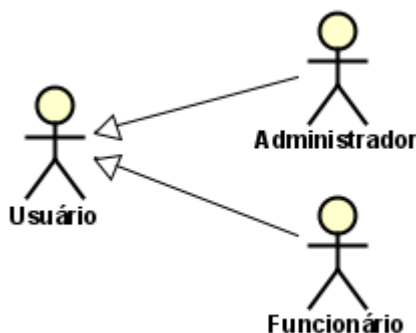
Os diagramas de caso de uso são uma ferramenta de modelagem de sistemas que descreve as interações entre os usuários e o sistema. Eles são uma forma de representar os requisitos funcionais de um sistema, mostrando as principais funcionalidades que o sistema deve fornecer e como os usuários interagem com essas funcionalidades.

Os diagramas de caso de uso são compostos por três elementos principais:

- Atores: Representam os usuários ou outros sistemas que interagem com o sistema.
- Casos de uso: Representam as funcionalidades do sistema que são relevantes para os atores.
- Relações entre atores e casos de uso: Representam como os atores interagem com os casos de uso.

Na figura abaixo, o ator Usuário foi generalizado em dois tipos de atores: Funcionário e Administrador. Essa generalização é feita com base nas responsabilidades e privilégios de cada tipo de ator. Como o Administrador pode realizar as mesmas funções de trabalho que o Funcionário, a única diferença está nas permissões de edição de usuários, onde o Funcionário só poderá editar o seu perfil pessoal. Já o Administrador poderá editar a si mesmo e os funcionários.

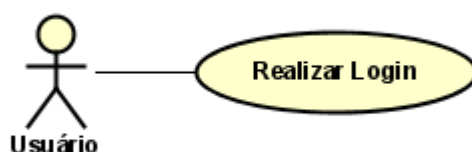
Figura 1 – Generalização do usuário.



Fonte: Autoria própria (2023).

Antes de acessar as principais funções, o usuário realizará o processo de login. Onde será obrigatório passar por esse processo de autenticação antes de ter acesso às demais funções do sistema. Este caso de uso foi separado dos outros para não gerar ambiguidades, mas corresponde a uma sequência lógica.

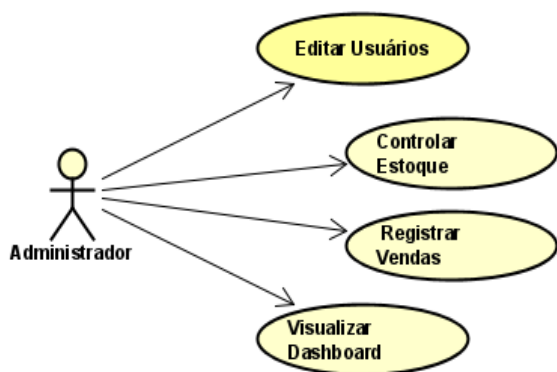
Figura 2 – Caso de Uso: Realizar Login



Fonte: Autoria própria (2023).

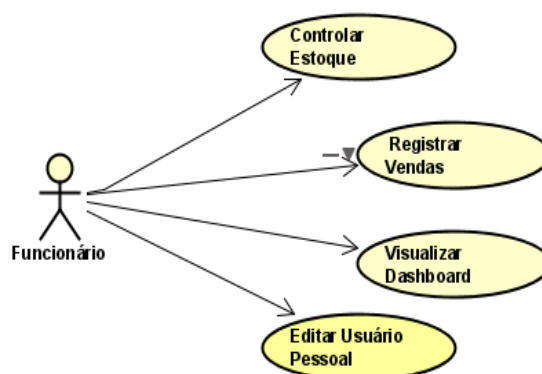
Como supracitado a única diferença entre os acessos do funcionário e do administrador é a permissão deste último ator de alterar não apenas o seu usuário, mas também os perfis dos demais usuários como pode ser observado nas figuras 3 e 4.

Figura 3 – Visão do Administrador



Fonte: Autoria própria (2023).

Figura 4 – Visão do Funcionário



Fonte: Autoria própria (2023).

4.2 Descrição dos Casos de Uso

A descrição de casos de uso é uma documentação que descreve um caso de uso de forma detalhada. Ela fornece informações sobre o objetivo do caso de uso, os atores envolvidos, as pré-condições, as pós-condições, o fluxo principal e os fluxos alternativos.

4.2.1 Realizar Login

Nome do Caso de Uso: Realizar Login

Atores: Administrador e Funcionário

Objetivo: Permitir que um usuário autentique sua identidade no sistema, fornecendo credenciais válidas, para obter acesso aos recursos e funcionalidades disponíveis.

Pré-condições:

- O sistema está em execução.
- O usuário possui credenciais válidas (registro e senha).

Pós-condições:

- O usuário está autenticado no sistema e tem acesso às funcionalidades autorizadas.

Fluxo Principal:

1. O usuário acessa a página de login do sistema.
2. O sistema apresenta os campos de entrada para nome de usuário e senha.
3. O usuário insere seu nome de usuário e senha.
4. O sistema valida as credenciais fornecidas.
5. Se as credenciais são válidas, o sistema autentica o usuário e concede acesso.
6. O sistema redireciona o usuário para a página inicial ou a área restrita, dependendo do nível de permissões.
7. O caso de uso é encerrado.

Fluxos Alternativos:

- Credenciais Inválidas (Passo 4):
 - Se as credenciais fornecidas não são válidas, o sistema exibe uma mensagem de erro e retorna ao passo 2, permitindo que o usuário tente novamente.
- Recuperação de Senha (Após Passo 4):
 - Se o usuário esqueceu sua senha, ele pode solicitar uma redefinição de senha.
 - Onde apenas um administrador poderá validar essa alteração.

4.2.2 Controlar Estoque

Nome do Caso de Uso: Controlar Estoque

Atores: Administrador e Funcionário

Objetivo: Permitir que os usuários realizem operações de controle de estoque, como adição, edição, remoção e visualização de produtos.

Pré-condições:

- O sistema está em execução.
- O usuário está logado

Pós-condições:

- As alterações no estoque são refletidas no sistema.
- Os usuários podem visualizar o estado atualizado do estoque.

Fluxo Principal:

1. O Funcionário ou Administrador acessa a funcionalidade de controle de estoque no sistema.
2. O sistema exibe as opções para adicionar, editar, remover ou visualizar produtos.
3. Adicionar Produto:
 - O usuário insere as informações do novo produto.
 - O sistema adiciona o novo produto ao estoque.
4. Editar Produto:
 - O usuário seleciona um produto existente.
 - O sistema exibe os detalhes atuais do produto.

- O usuário faz as alterações necessárias nos detalhes do produto.
- O sistema atualiza as informações do produto no estoque.

5. Remover Produto:

- O usuário seleciona um produto existente.
- O sistema confirma a remoção do produto.
- O sistema remove o produto do estoque.

6. Visualizar Produtos:

- O sistema exibe a lista de todos os produtos no estoque, incluindo detalhes como nome, quantidade disponível e preço.

Fluxos Alternativos:

- Produto Já Existente (Adicionar Produto - Passo 3):
 - Se o nome do produto já estiver no estoque, o sistema exibe uma mensagem indicando que o produto já existe.
- Produtos Indisponíveis (Visualizar Produtos - Passo 6):
 - Se não houver produtos no estoque, o sistema exibe uma mensagem indicando que o estoque está vazio.

4.2.3 Visualizar Dashboard

Nome do Caso de Uso: Visualizar Dashboard

Atores: Administrador e Funcionário

Objetivo: Permitir que o Funcionário e o Administrador visualizem um dashboard com informações relevantes sobre o desempenho e as métricas do mercadinho.

Pré-condições:

- O sistema está em execução.
- O usuário está logado

Pós-condições:

- O Funcionário e o Administrador podem visualizar as informações do dashboard.

Fluxo Principal:

1. O usuário acessa a funcionalidade de visualização do dashboard no sistema.
2. O sistema exibe gráficos e estatísticas relacionadas ao desempenho do mercadinho.

Fluxos Alternativos:

- Informação Indisponível (Passo 2):
 - O usuário tenta acessar informações que não estão disponíveis.

4.2.4 Registrar Vendas

Nome do Caso de Uso: Registrar Vendas

Atores: Administrador e Funcionário

Objetivo: Permitir que o Funcionário ou Administrador registre vendas no sistema, inserindo informações sobre os produtos vendidos.

Pré-condições:

- O sistema está em execução.
- O usuário está logado

Pós-condições:

- A venda é registrada no sistema.
- O estoque é atualizado com as quantidades vendidas.

Fluxo Principal:

1. O Funcionário ou Administrador acessa a funcionalidade de registro de vendas no sistema.
2. O sistema exibe uma interface para inserção de informações sobre a venda.
3. O usuário insere o ID do produto vendido.
4. O sistema verifica a existência do produto no estoque.
5. Se o produto existe, o sistema exibe as informações do produto.
6. O usuário insere a quantidade vendida.
7. O sistema calcula o valor total da venda.

8. O usuário confirma a venda.
9. O sistema registra a venda no histórico de transações.
10. O sistema atualiza o estoque subtraindo a quantidade vendida dos produtos.

Fluxos Alternativos:

- Produto não Encontrado (Passo 5):
 - Se o produto não for encontrado no estoque, o sistema exibe uma mensagem informando que o produto não está disponível para venda.
- Estoque Insuficiente (Passo 6):
 - Se a quantidade solicitada for maior do que a disponível em estoque, o sistema exibe uma mensagem indicando que não há estoque suficiente para a venda.
- Cancelamento da Venda (Passo 8):
 - O usuário pode optar por cancelar a venda a qualquer momento antes de confirmar. Nesse caso, a venda não é registrada, e o estoque não é atualizado.

4.2.5 Editar Usuário Pessoal

Nome do Caso de Uso: Editar Usuário Pessoal

Ator: Funcionário

Objetivo: Permitir que o Funcionário edite suas informações pessoais no sistema.

Pré-condições:

- O sistema está em execução.
- O Funcionário deve estar logado.

Pós-condições:

- As informações pessoais do Funcionário são atualizadas no sistema.

Fluxo Principal:

1. O Funcionário acessa a funcionalidade de edição de usuário pessoal no sistema.
2. O sistema exibe as informações pessoais atuais do Funcionário.
3. O Funcionário faz as alterações desejadas nos campos relevantes.

4. O Funcionário confirma as alterações.
5. O sistema valida as informações atualizadas.
6. Se as informações são válidas, o sistema atualiza o perfil do Funcionário.
7. O sistema exibe uma mensagem de sucesso.
8. O caso de uso é encerrado.

Fluxos Alternativos:

- Validação de Informações (Passo 5):
 - Se as informações inseridas não atenderem aos critérios de validação, o sistema exibe uma mensagem de erro e retorna ao Passo 2.

4.2.6 Editar Usuários

Nome do Caso de Uso: Editar Usuários

Ator: Administrador

Objetivo: Permitir que o Administrador edite informações pessoais de usuários no sistema, incluindo o próprio perfil.

Pré-condições:

- O sistema está em execução.
- O Administrador está autenticado no sistema.

Pós-condições:

- As informações pessoais do usuário são atualizadas no sistema.

Fluxo Principal:

1. O Administrador acessa a funcionalidade de edição de usuários no sistema.
2. O sistema exibe uma lista de usuários cadastrados.
3. O Administrador seleciona o usuário que deseja editar.
4. O sistema exibe as informações pessoais atuais do usuário selecionado.
5. O Administrador faz as alterações desejadas nos campos relevantes.
6. O Administrador confirma as alterações.
7. O sistema valida as informações atualizadas.

8. Se as informações são válidas, o sistema atualiza o perfil do usuário.

9. O sistema exibe uma mensagem de sucesso.

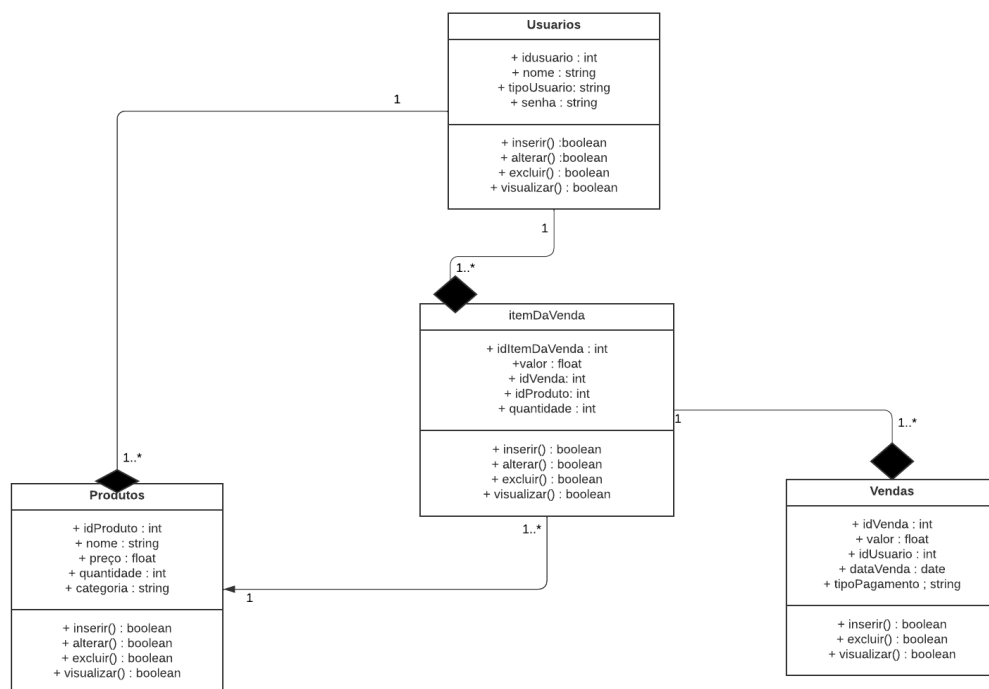
Fluxos Alternativos:

- Validação de Informações (Passo 7):
 - Se as informações inseridas não atenderem aos critérios de validação, o sistema exibe uma mensagem de erro e retorna ao Passo 4.

5. DIAGRAMA DE CLASSE

O diagrama de classes é uma representação visual que é de suma importância para o desenvolvimento de software, a qual tem o objetivo de estruturar e organizar os objetos essenciais do sistema.

Figura 5 – Diagrama de classes.



Fonte: Autoria própria (2023).

Esse diagrama de classes representa a estrutura de dados básica para um sistema de vendas, com ênfase em usuários, produtos, vendas e itens de venda. Aqui está um resumo:

- Classe Usuarios:

1. Atributos: idusuario (identificação), nome, tipoUsuario (como "Funcionário" ou "Administrador"), senha.
2. Métodos: inserir(), alterar(), excluir(), visualizar() para operações relacionadas aos usuários.
3. Associação: Relacionamento de 1 para muitos entre a classe Usuarios e Vendas, indicando que um usuário pode realizar várias vendas.

- Classe Produtos:

1. Atributos: idProduto (identificação), nome, preço, quantidade, categoria.
2. Métodos: inserir(), alterar(), excluir(), visualizar() para operações relacionadas aos produtos.
3. Associação: Relacionamento de 1 para muitos entre a classe Produtos e itemDaVenda, indicando que um produto pode estar presente em vários itens de venda.

- Classe itemDaVenda:

1. Atributos: idItemDaVenda (identificação), valor, idVenda (referência à venda associada), idProduto (referência ao produto associado), quantidade.
2. Métodos: inserir(), alterar(), excluir(), visualizar() para operações relacionadas aos itens de venda.
3. Associação: Relacionamento de 1 para muitos entre a classe itemDaVenda e Vendas, indicando que um item de venda pertence a uma venda específica.

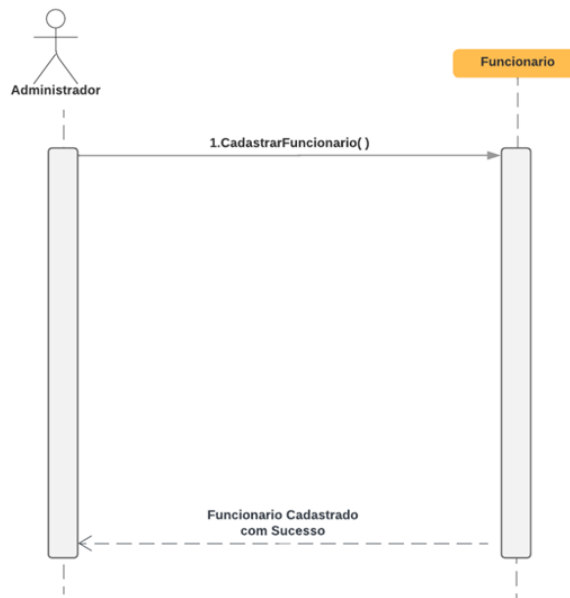
- Classe Vendas:

1. Atributos: idVenda (identificação), valor, idUsuario (referência ao usuário associado), dataVenda, tipoPagamento.
2. Métodos: inserir(), excluir(), visualizar() para operações relacionadas às vendas.

6. DIAGRAMA DE SEQUÊNCIA

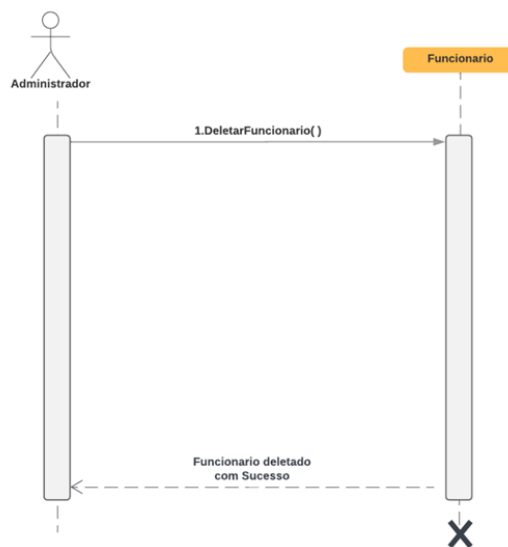
O diagrama de sequência apresenta o fluxo detalhado das interações entre objetos no processo de cadastro de usuário e de produto. Além das interações entre produto e venda.

Figura 6 – Diagrama de sequência para a classe funcionário.



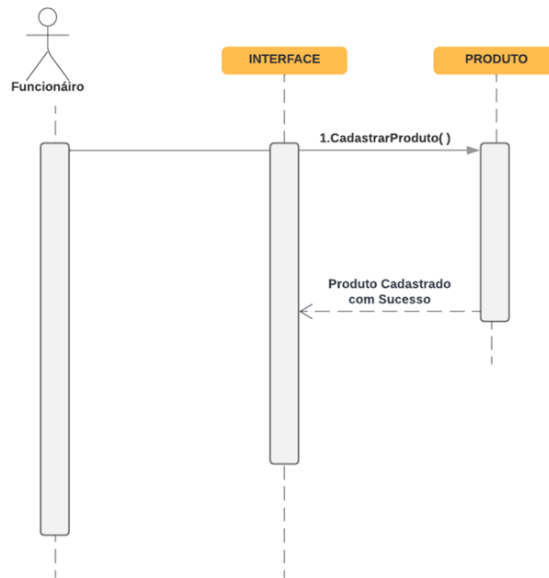
Fonte: Autoria própria (2023).

Figura 7 – Diagrama de sequência para a classe funcionario. Usando o método `DeletarFuncionario()`.



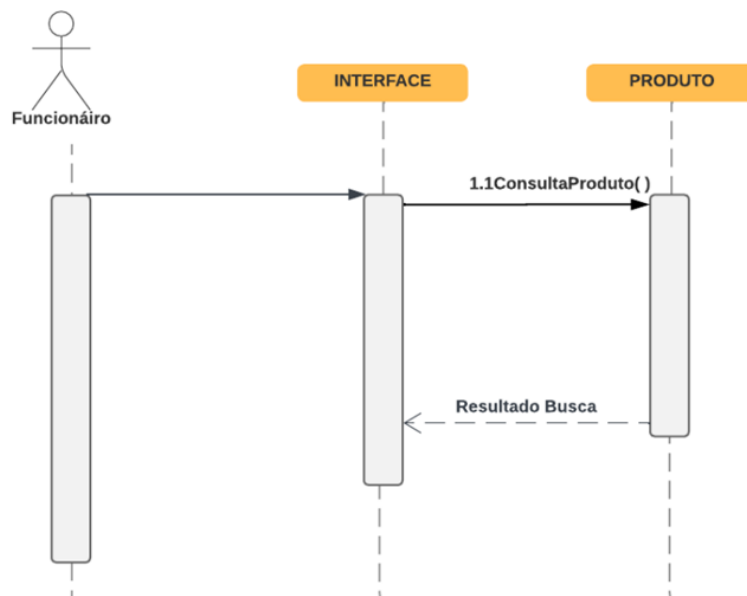
Fonte: Autoria própria (2023).

Figura 8 – Diagrama de sequência para a classe produto. Usando o método cadastrarProduto().



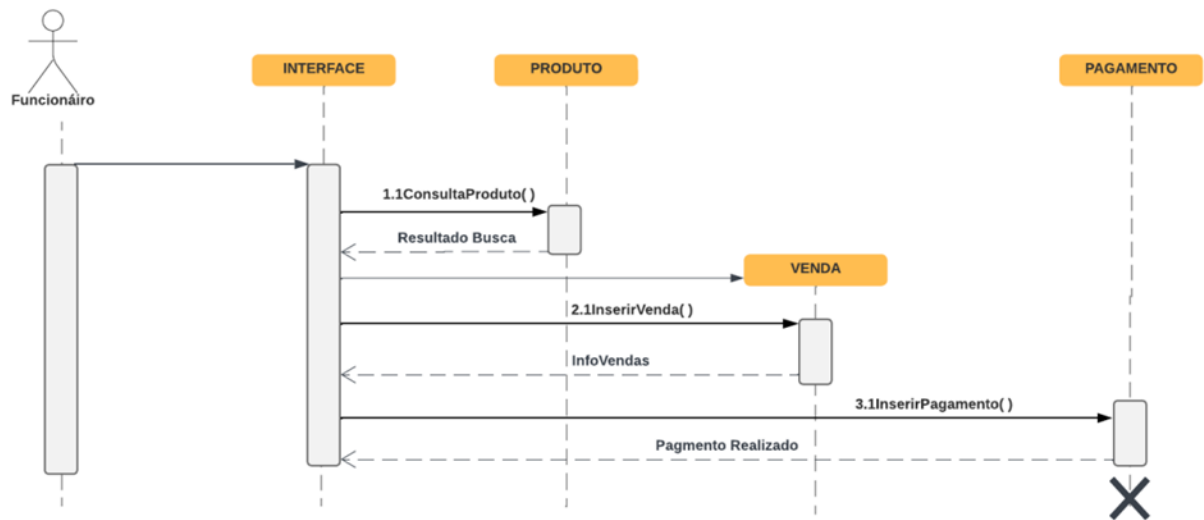
Fonte: Autoria própria (2023).

Figura 9 – Diagrama de sequência para a classe produto. Usando o método ConsultaProduto().



Fonte: Autoria própria (2023).

Figura 10 – Diagrama de sequência do sistema. Diagramando o processo de consulta até venda do produto.



Fonte: Autoria própria (2023).

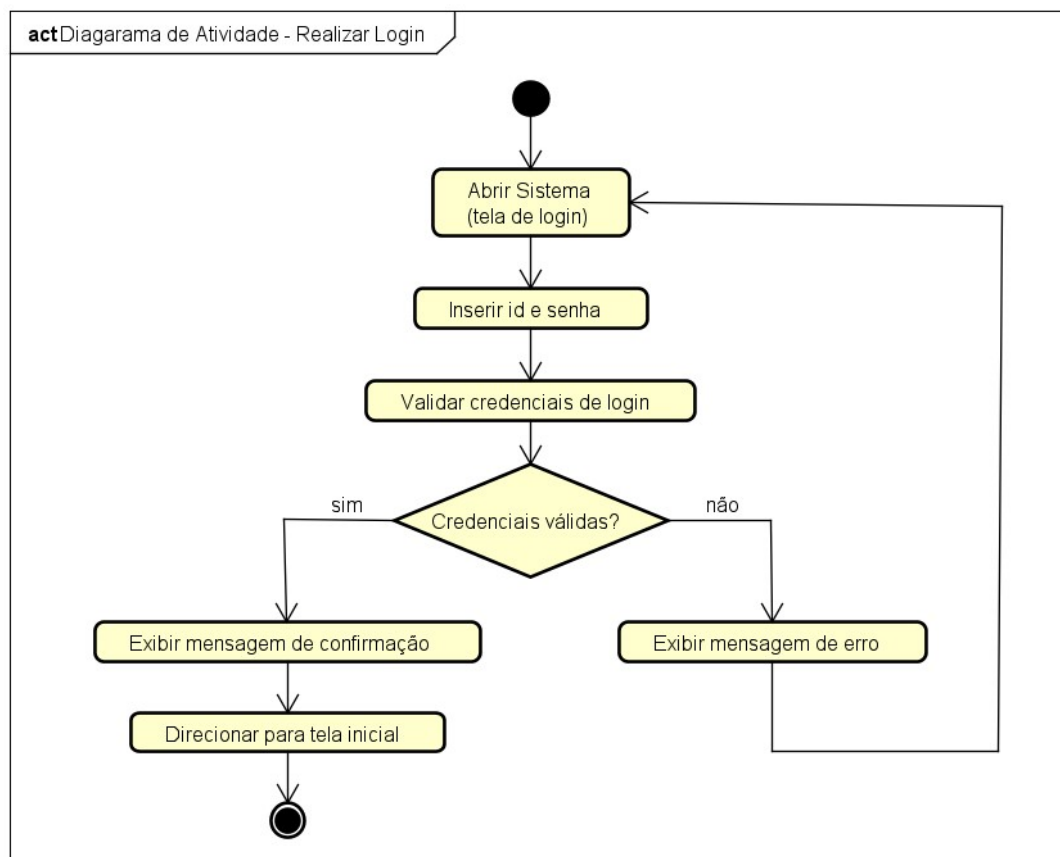
7. DIAGRAMA DE ATIVIDADE

O diagrama de atividade tem por objetivo fornecer uma visão clara e compreensível do fluxo de controle no sistema, como um mapa, ajudando os desenvolvedores a entenderem a lógica do processo e facilitando a comunicação entre membros da equipe.

Os diagramas de atividade produzidos estão apresentados a seguir, sendo cada atividade definida por uma unidade de trabalho, nesse contexto, os casos de uso definidos no tópico 4 desse documento.

7.1 Realizar Login

Figura 11 – Diagrama da atividade “Realizar Login”.

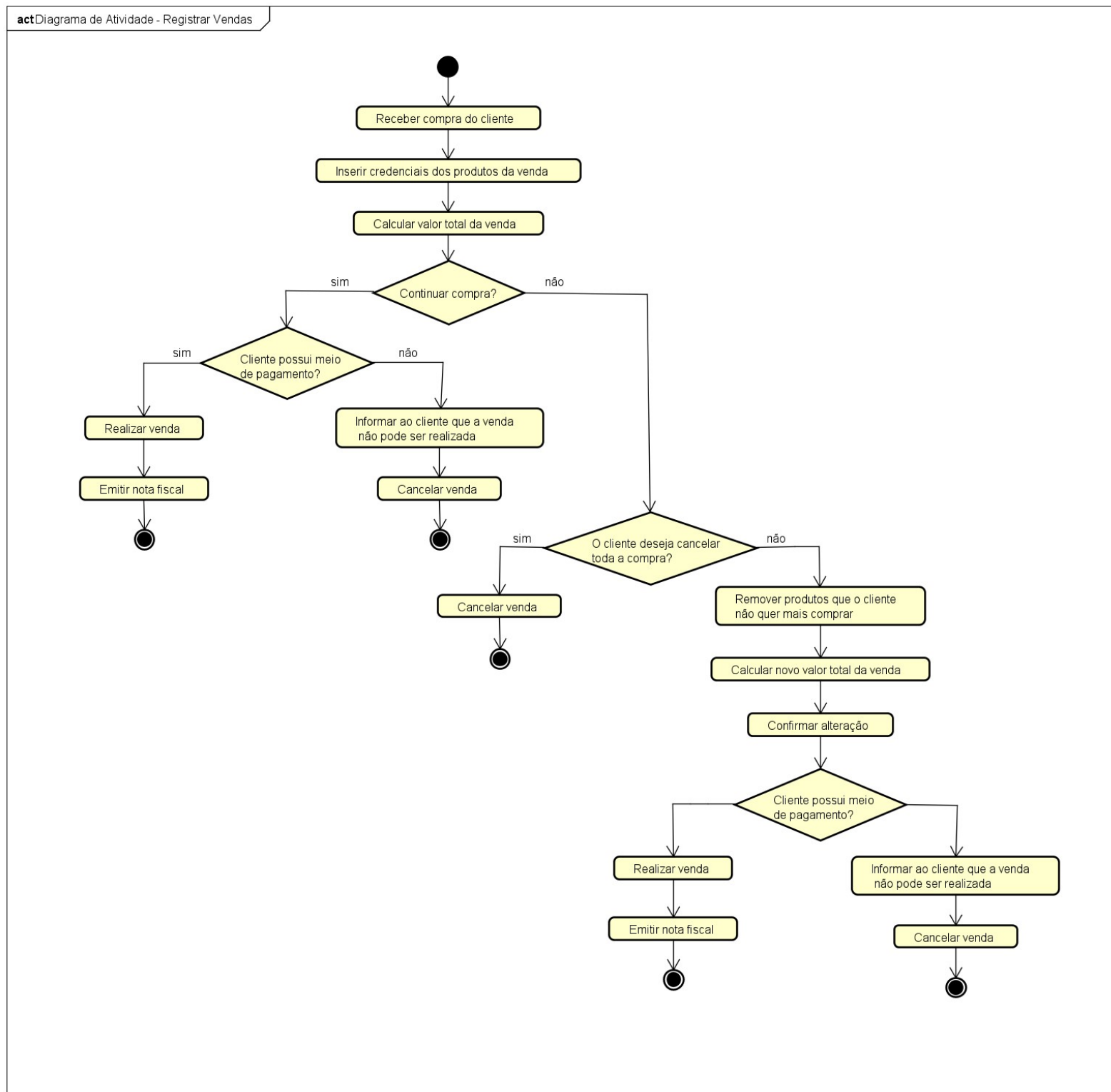


Fonte: Autoria Própria (2023).

Este diagrama de atividade ilustra o fluxo de atividades para realizar o login em um sistema. O processo começa quando o usuário abre o sistema e insere seu nome de usuário e senha. O sistema então valida as credenciais do usuário. Se as credenciais forem válidas, o usuário é direcionado para a tela principal do sistema. Se as credenciais estiverem inválidas, o usuário recebe uma mensagem de erro e é redirecionado para a tela de login.

7.2 Registrar Vendas

Figura 12 – Diagrama da atividade “Registrar Vendas”.

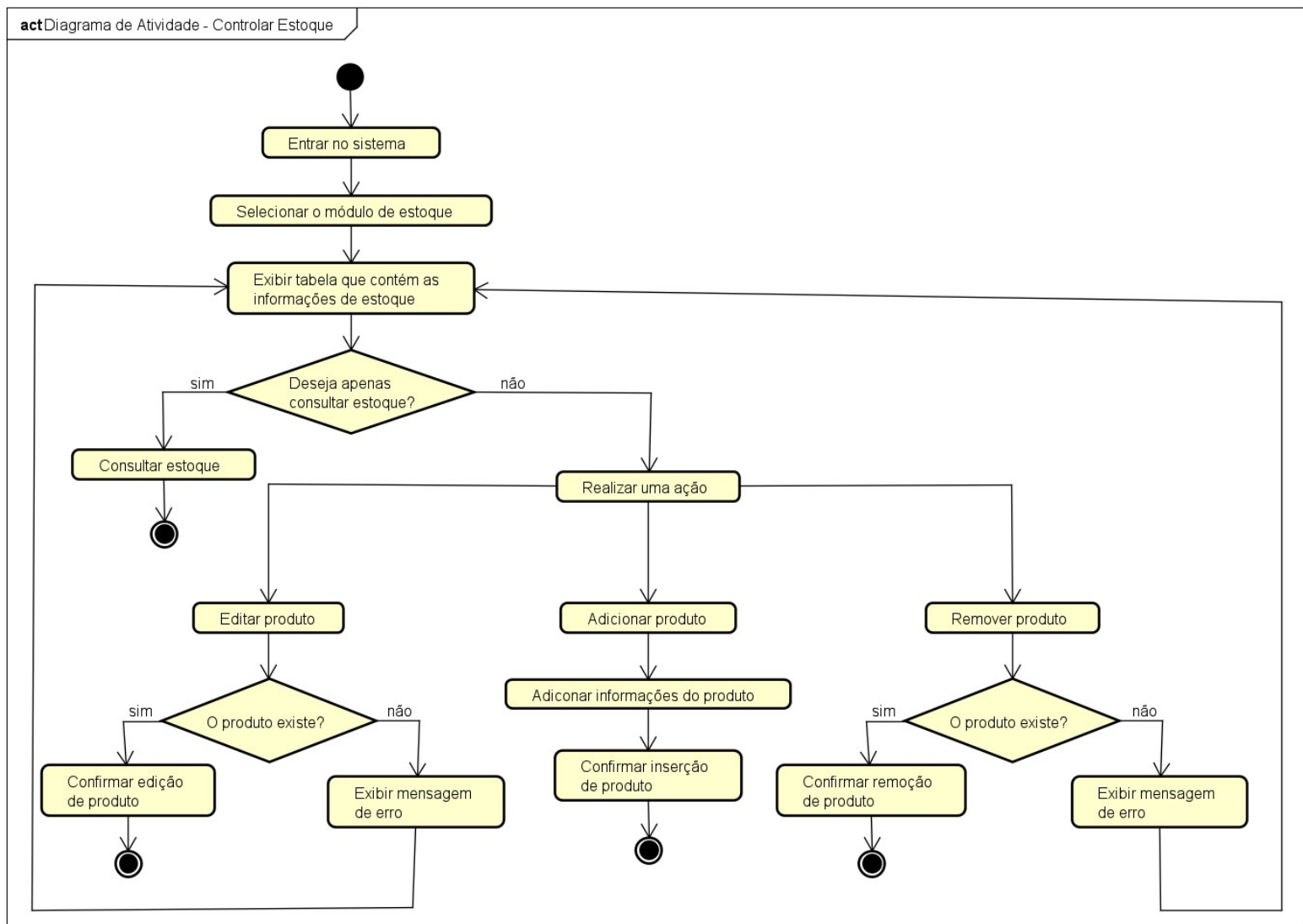


Fonte: Autoria Própria (2023).

O diagrama de atividade da figura acima ilustra o fluxo de atividades para registrar uma venda em um sistema de comércio eletrônico. O processo começa quando o cliente vai ao caixa realizar a sua compra. Em seguida, o usuário, já logado no sistema, insere as credenciais dos produtos que compõem a venda. Após isso, o sistema calcula o valor total da venda. Se o cliente desejar continuar com a compra e obtiver os meios necessários para pagamento, o sistema processa a venda e emite nota fiscal ao cliente. Se não, o funcionário ou administrador informa que a venda não pode ser realizada.

7.3 Controlar Estoque

Figura 13 – Diagrama da atividade “Controlar Estoque”.

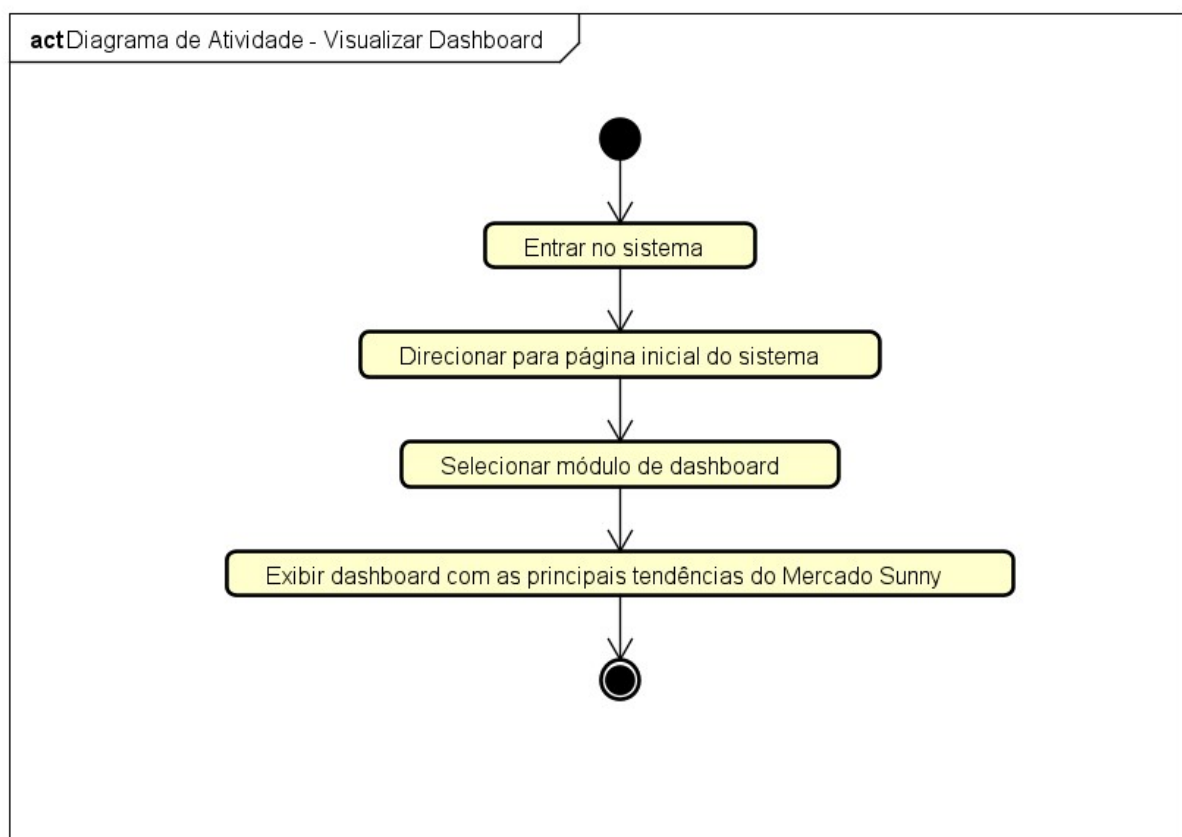


Fonte: Autoria Própria (2023).

O diagrama de atividade representado pela figura acima ilustra o fluxo de atividades para controlar o estoque em um sistema. O processo começa quando o usuário entra no sistema e seleciona o módulo de estoque. O sistema então exibe as informações de estoque atuais. O usuário pode então solicitar a realização de uma ação, como apenas consultar o estoque, editar um produto, remover um produto ou adicionar um produto.

7.4 Visualizar Dashboard

Figura 14 – Diagrama da atividade “Visualizar Dashboard”.

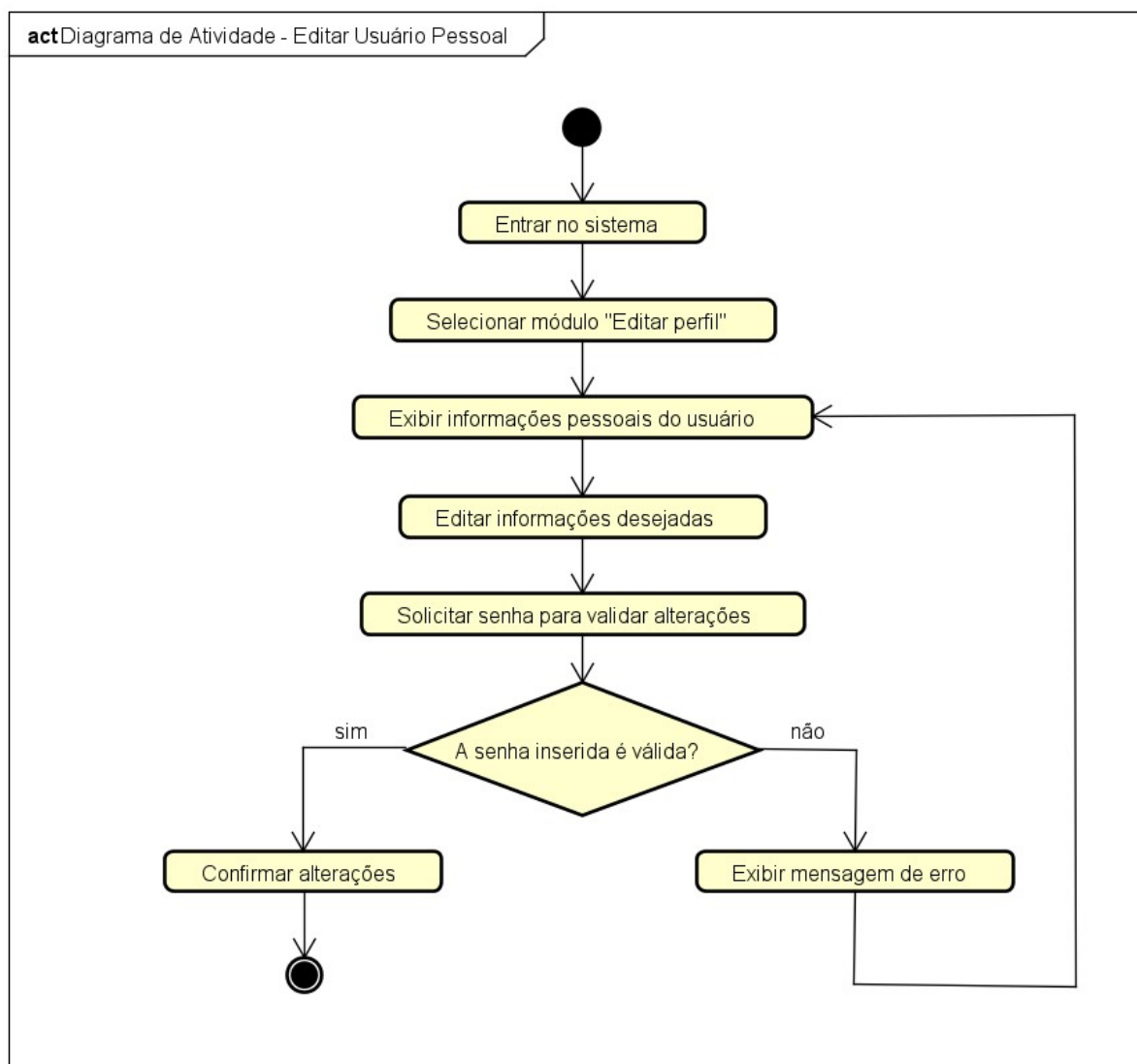


Fonte: Autoria Própria (2023).

O diagrama de atividade representado pela figura acima ilustra o fluxo de atividades para visualizar um dashboard em um sistema. O processo começa quando o usuário entra no sistema e seleciona o módulo de dashboard. O sistema então exibe o dashboard com as informações relevantes.

7.5 Editar Usuário Pessoal

Figura 15 – Diagrama da atividade “Editar Usuário Pessoal”.

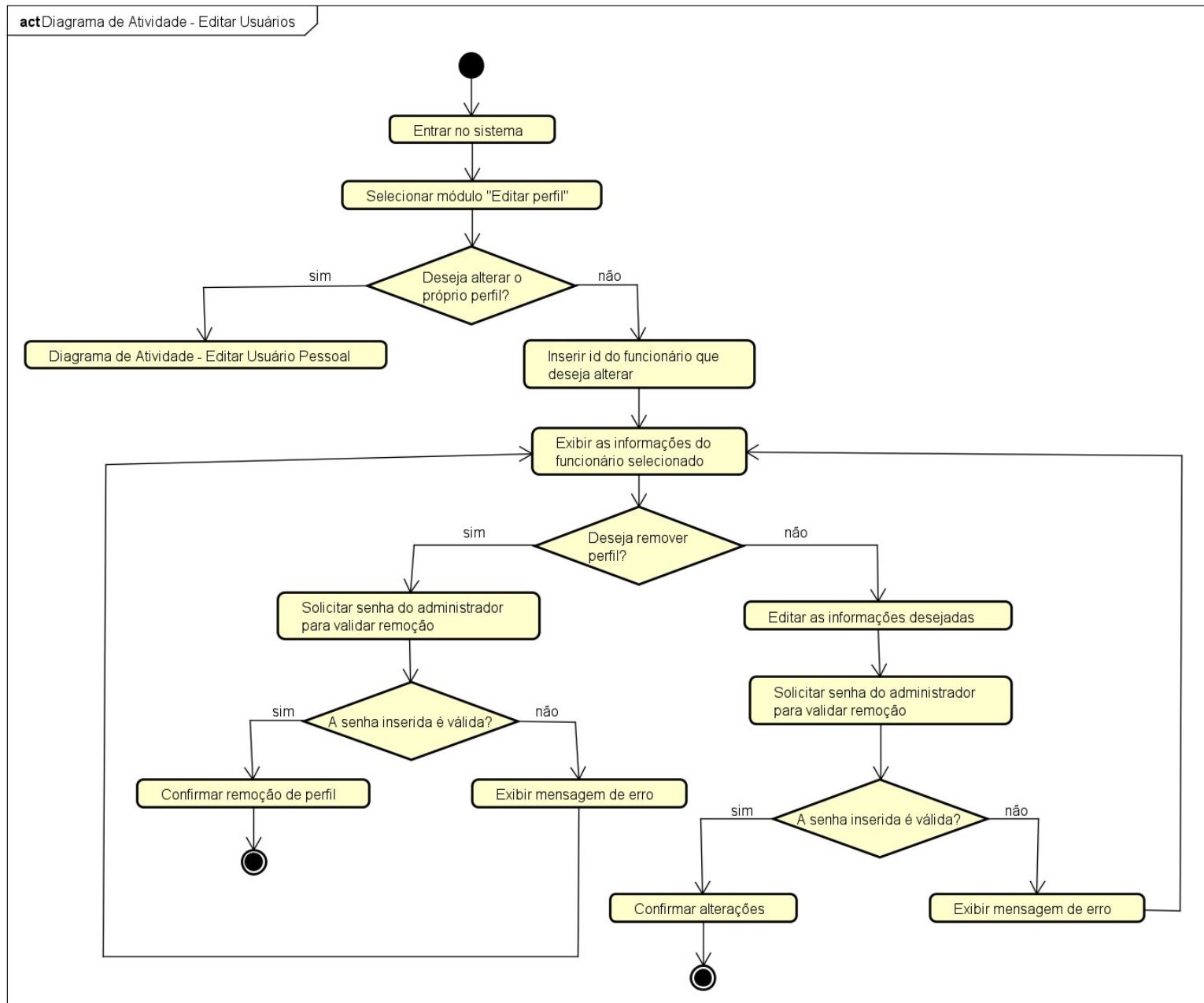


Fonte: Autoria Própria (2023).

O diagrama de atividade representado pela figura acima ilustra o fluxo de atividades para editar as informações pessoais de um usuário em um sistema. O processo começa quando o usuário entra no sistema e seleciona o módulo de "Editar perfil". O sistema então exibe as informações pessoais do usuário. Se a senha do usuário for válida, o sistema confirma as alterações feitas. Se não, o sistema exibe uma mensagem de erro e o usuário é redirecionado para a página de edição.

7.6 Editar Usuários

Figura 16 – Diagrama da atividade “Editar Usuários”.



Fonte: Autoria Própria (2023).

O diagrama de atividade representado pela figura acima ilustra o fluxo de atividades para editar as informações pessoais de um usuário em um sistema. O processo começa quando o usuário entra no sistema e seleciona o módulo de "Editar perfil". O sistema exibe as informações pessoais de todos os usuários, isto é, ele pode alterar as informações dele e dos outros. Se a senha do usuário for válida, o sistema confirma as alterações feitas. Se não, o sistema exibe uma mensagem de erro e o usuário é redirecionado para a página de edição.

8. TECNOLOGIAS USADAS

O programa apresentado neste documento foi desenvolvido utilizando PHP e o framework front-end Bootstrap, que integra JavaScript e HTML na construção da interface. O código foi elaborado no IDE e compilador do Microsoft Visual Studio Code, com o framework servindo como a base da estrutura de CSS do sistema. Devido à adoção do padrão PHP, é possível recompilá-lo nos compiladores MinGW (Windows), GCC (Linux) e CLang (macOS), proporcionando versatilidade de execução em diferentes ambientes operacionais. Para a alimentação dos dados de administradores, funcionários e produtos do mercadinho Sunny, foi utilizado o MySQL e o HeidiSQL para a sua administração.

Além disso, empregou-se o Astah para criar os diagramas de caso de uso, de classe, de sequência e de atividade, enriquecendo a documentação do projeto com representações visuais abrangentes e facilitando a compreensão do sistema em diferentes níveis de detalhe.

Tabela 1 – Tecnologias usadas.

Tecnologia	Especificação	Função
PHP	Linguagem de programação server-side	Desenvolvimento web, scripts do lado do servidor
Bootstrap	Framework front-end para desenvolvimento web	Desenvolvimento responsivo, design moderno
Apex Charts	Biblioteca de gráficos interativos em JavaScript	Visualização de dados, dashboards
Microsoft Visual Studio Code	Ambiente de desenvolvimento integrado (IDE)	Desenvolvimento de software em várias linguagens
MySQL	Sistema de gerenciamento de banco de dados relacional	Armazenamento e recuperação de dados
HeidiSQL	Ferramenta de administração para bancos de dados MySQL	Gerenciamento e manipulação de bancos de dados
Astah	Ferramenta de modelagem UML	Modelagem de sistemas, diagramas UML

JavaScript	Linguagem de programação client-side	Desenvolvimento web interativo, manipulação de DOM
HTML	Linguagem de marcação para construção de páginas web	Estruturação de conteúdo web
CSS	Linguagem de estilo para design de páginas web	Estilo e aparência de páginas web

Fonte: Autoria Própria (2023).

9. ESCOPO DA PROTOTIPAÇÃO

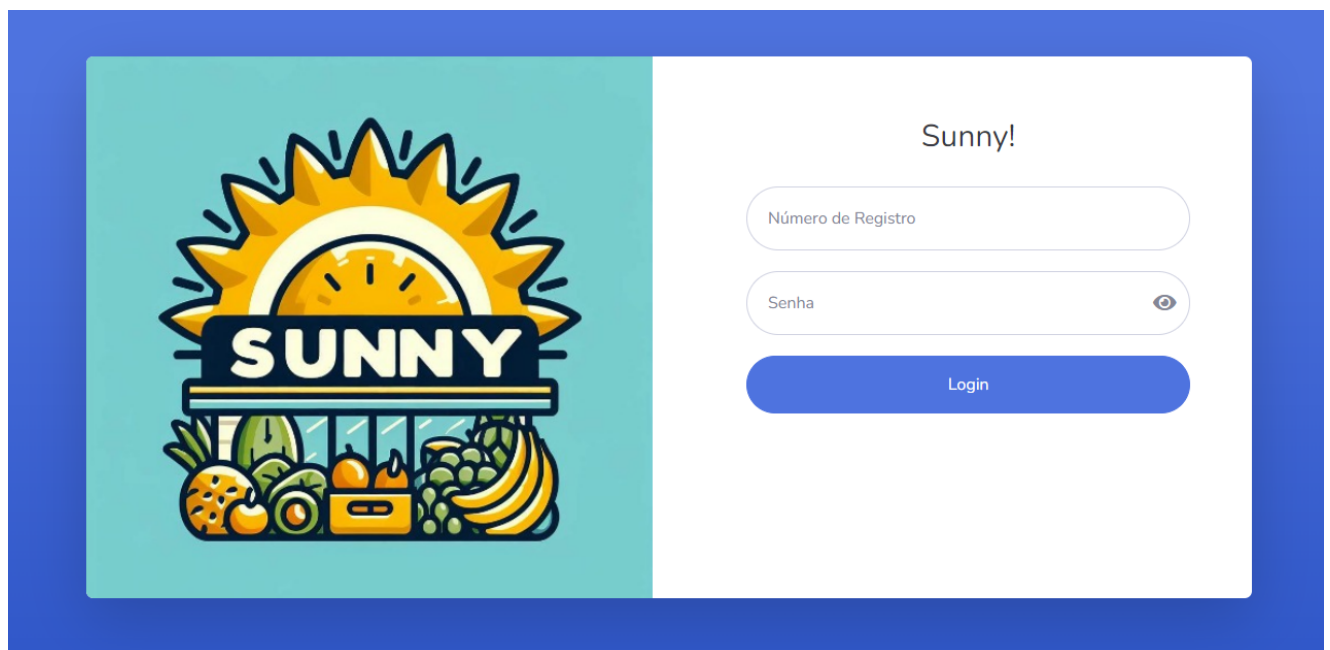
Além dos diagramas apresentados, o processo de desenvolvimento desse sistema para o mercadinho Sunny também contou com a prototipação da aplicação web. Este protótipo visual e funcional oferece uma visão tangível das características e da experiência do usuário que estão sendo projetadas. Ao explorar este protótipo, é possível mergulhar nas interfaces, navegação e funcionalidades previstas, permitindo uma compreensão mais aprofundada do design antes da implementação completa. Ao ler este capítulo, será possível visualizar os principais elementos do software e saber como eles contribuem para o projeto final.

Para o escopo da prototipação, foram desenvolvidas as funcionalidades enumeradas a seguir.

9.1 Login

Na tela de login apresentada abaixo, o usuário, seja ele administrador ou funcionário, é capaz de escrever as credenciais necessárias para acessar o sistema do Mercado Sunny. Ademais, é notório que o usuário pode visualizar a sua senha ao clicar no ícone à direita da caixa “Senha”, para não correr o risco de digitá-la incorretamente. Essa funcionalidade visa evitar equívocos na digitação e proporcionar maior segurança ao usuário, permitindo uma verificação visual antes da submissão das informações de login.

Figura 17 – Tela de login.

A imagem mostra a interface de login do sistema Sunny. O layout é dividido em duas seções principais. À esquerda, há um cartão com um fundo verde-água que contém o logo 'SUNNY' em letras brancas e negritadas, sobreposto a uma ilustração de um sol amarelo sorridente e uma variedade de frutas frescas (laranja, maçã, banana, uva, melão) e vegetais (brócolis, tomate). À direita, o fundo é branco e contém o texto 'Sunny!' no topo. Abaixo dele, há dois campos de entrada: 'Número de Registro' e 'Senha'. O campo 'Senha' possui um ícone de olho à direita, permitindo alternar a visibilidade da senha. Abaixo dos campos, há um botão azul com o texto 'Login' em branco.

Fonte: Autoria Própria (2023).

9.2 Dashboard

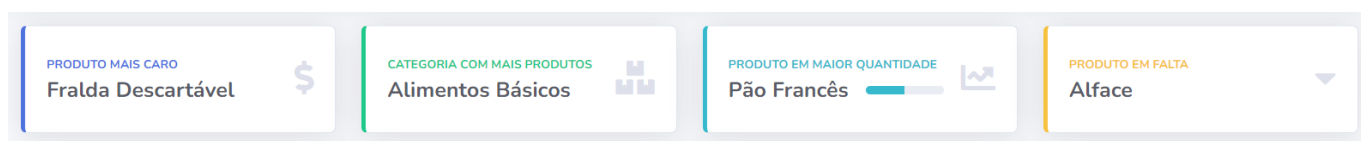
Ao efetuar o login, o usuário é instantaneamente conduzido à tela inicial da aplicação do Mercadinho Sunny. Neste ponto, encontra-se uma visão abrangente através de cards e do primeiro dashboard, que apresenta informações cruciais sobre o estabelecimento. Vale ressaltar que este é apenas o ponto de partida, pois a equipe de desenvolvimento está em constante aprimoramento, planejando a implementação de diversos dashboards adicionais.

A funcionalidade principal dessa tela reside na capacidade do usuário em visualizar de maneira intuitiva informações e tendências preponderantes no mercado. Este recurso inclui, entre outros, a contagem de produtos categorizados por estoque, proporcionando uma visão detalhada e organizada do inventário. Tal abordagem não apenas facilita a compreensão, mas também agiliza a tomada de decisões estratégicas para a gestão do estabelecimento.

Um diferencial relevante é a opção de efetuar o download dos gráficos apresentados nos formatos SVG, PNG e CSV. Esse recurso, acessível por meio de um ícone no canto superior direito do painel, possibilita ao usuário a obtenção de dados consolidados para análises mais aprofundadas, compartilhamento com outros membros da equipe ou arquivamento para referências futuras.

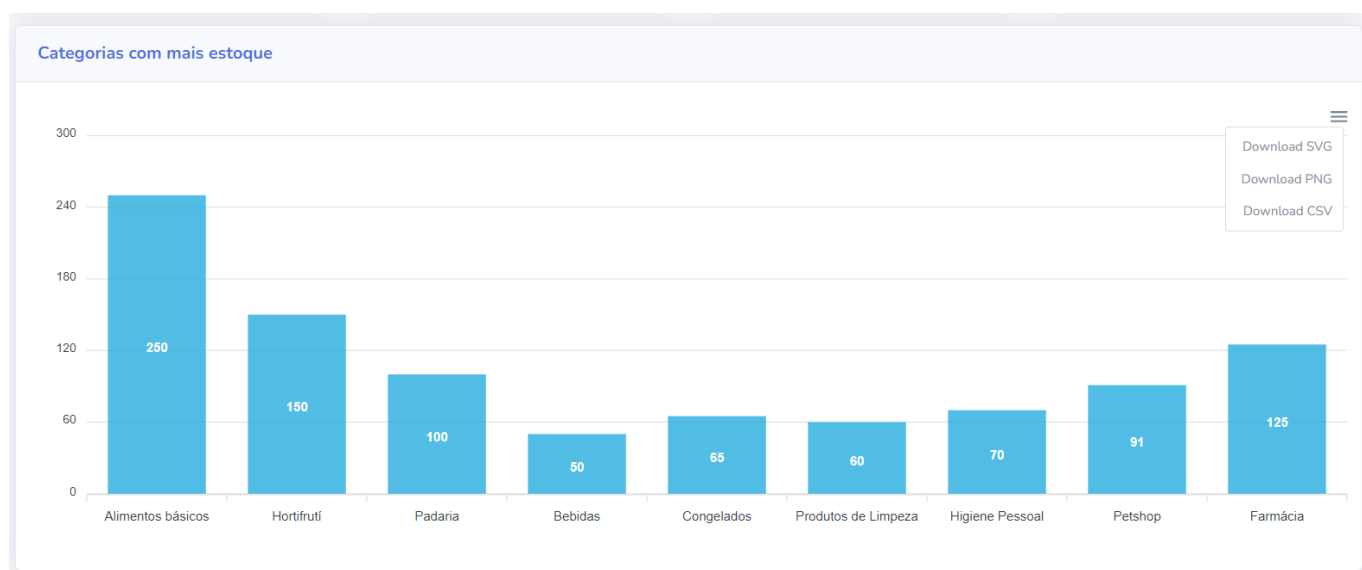
Adicionalmente, a aplicação é projetada para se adaptar dinamicamente às demandas do mercado, incorporando futuras funcionalidades e atualizações, mantendo-se alinhada com as necessidades em constante evolução do Mercadinho Sunny.

Figura 18 – Cards da página de dashboard



Fonte: Autoria Própria (2023).

Figura 19 – Dashboard com as informações de categorias de estoque.



Fonte: Autoria Própria (2023).

9.2 Estoque

Uma funcionalidade de extrema relevância para a gestão eficaz do mercado reside no controle de estoque, acessível ao usuário por meio da barra lateral do sistema. A tabela de estoque engloba todas as características dos produtos disponíveis na loja, como identificação (id), nome, categoria, quantidade, preço, marca e peso. Além desses dados essenciais, destaca-se a presença de uma coluna especial denominada "Ações", que desempenha um papel crucial na constante atualização das informações de estoque.

A mencionada coluna oferece duas funcionalidades indispensáveis: a edição e a remoção de produtos. Essas ferramentas são fundamentais para assegurar a precisão das informações, permitindo ajustes conforme necessário e a remoção de itens obsoletos ou fora de circulação. Trata-se de um mecanismo eficaz para manter o estoque alinhado com a dinâmica das demandas do mercado.

Outra função de destaque é a capacidade de adicionar produtos, disponível através de um botão no canto superior direito da tabela. Ao acionar essa funcionalidade, o usuário é conduzido a um formulário onde pode cadastrar todas as características de um novo produto, exceto o ID, que é atribuído automaticamente. Após a inclusão, o produto é imediatamente registrado no banco de dados da aplicação, assegurando uma atualização em tempo real e uma gestão eficiente do inventário. Essa agilidade no processo contribui para a operação fluida do estabelecimento, proporcionando uma visão detalhada e precisa do status do estoque.

Figura 20 – Tabela de Estoque.

Estoque

Gerenciamento

Adicionar Produto

Show

10

entries

Search:

ID	Nome	Categoria	Quantidade	Preço	Marca	Peso (Kg/L)	Ações
1	Arroz Branco	Alimentos Básicos	60	5.97	Tio José	1	<div><div></div><div></div></div>
2	Feijão Branco	Alimentos Básicos	30	4.5	Camil	1	<div><div></div><div></div></div>
3	Macarrão	Alimentos Básicos	40	2.99	Adria	0.5	<div><div></div><div></div></div>
4	Açúcar	Alimentos Básicos	25	3.75	União	1	<div><div></div><div></div></div>
5	Óleo de Soja	Alimentos Básicos	20	6.5	Liza	900	<div><div></div><div></div></div>
6	Feijão Preto	Alimentos Básicos	40	5.75	Camil	1.2	<div><div></div><div></div></div>
7	Macarrão Integral	Alimentos Básicos	25	3.49	Vitarella	0.6	<div><div></div><div></div></div>

Fonte: Autoria Própria (2023).

Figura 21 – Modal Editor Produto.

Estoque

Gerenciamento

Adicionar Produto

Show

10

entries

Search:

ID	Nome	Categoria	Quantidade	Preço	Marca	Peso (Kg/L)	Ações
1	Arroz Branco	Alimentos Básicos	60	5.97	Tio José	1	<div><div></div><div></div></div>
2	Feijão Branco	Alimentos Básicos	30	4.5	Camil	1	<div><div></div><div></div></div>
3	Macarrão	Alimentos Básicos	40	2.99	Adria	0.5	<div><div></div><div></div></div>
4	Açúcar	Alimentos Básicos	25	3.75	União	1	<div><div></div><div></div></div>
5	Óleo de Soja	Alimentos Básicos	20	6.5	Liza	900	<div><div></div><div></div></div>
6	Feijão Preto	Alimentos Básicos	40	5.75	Camil	1.2	<div><div></div><div></div></div>
7	Macarrão Integral	Alimentos Básicos	25	3.49	Vitarella	0.6	<div><div></div><div></div></div>

Editar Produto

Nome:

Arroz Branco

Categoria:

Alimentos Básicos

Quantidade:

60

Preço:

5.97

Marca:

Tio José

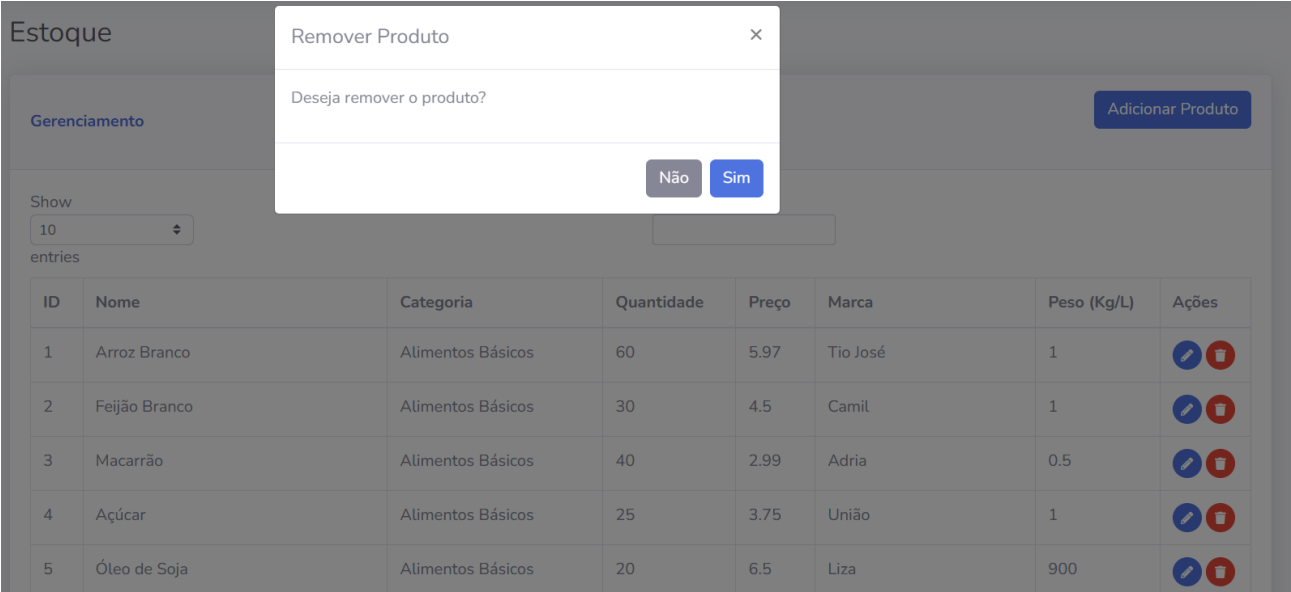
Peso (Kg/L):

1

Salvar Alterações

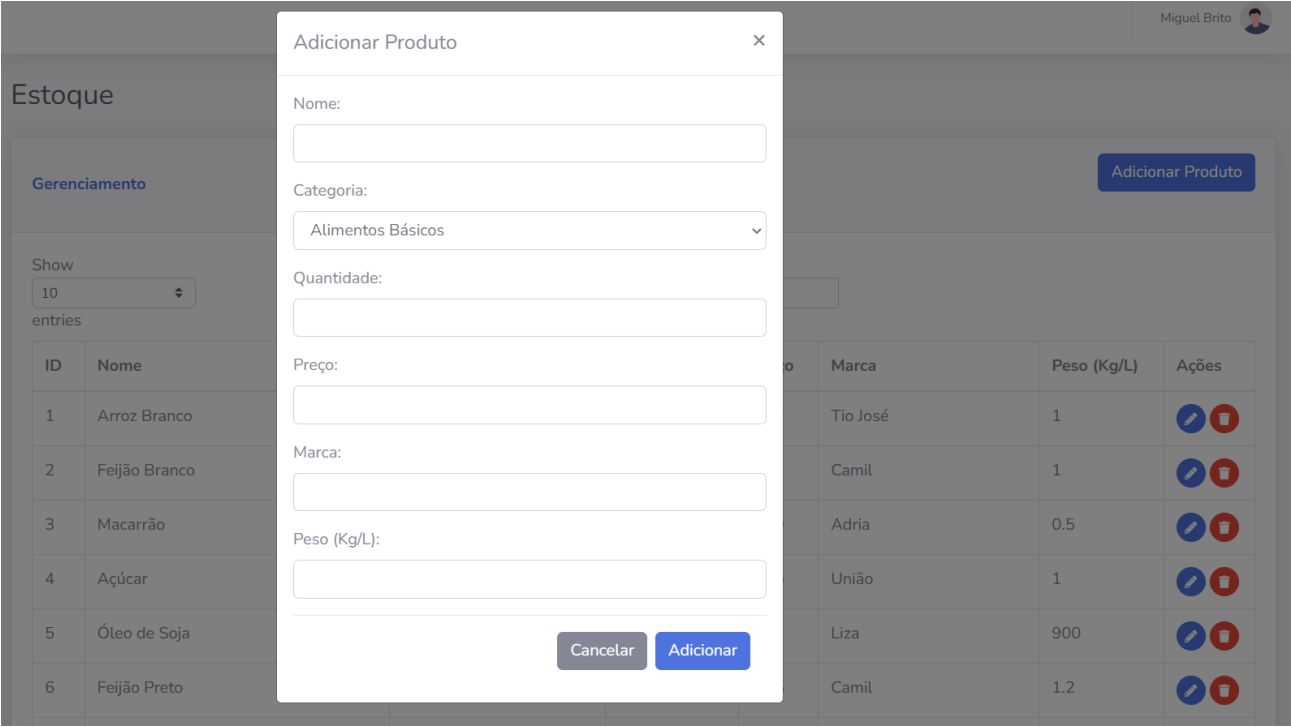
Fonte: Autoria Própria (2023).

Figura 22 – Modal Remover Produto.



Fonte: Autoria Própria (2023).

Figura 23 – Modal Adicionar Produto.



Fonte: Autoria Própria (2023).

9.2 Usuário

Por último, destaca-se a funcionalidade de usuário, na qual tanto o administrador quanto os funcionários do Mercado Sunny têm a capacidade de ajustar suas informações pessoais, como número de registro, nome e senha, acessando a tabela de configurações do usuário. Essas funcionalidades são compartilhadas entre o administrador e os funcionários, promovendo uma gestão eficaz da plataforma.

Como detentor de uma posição de liderança no mercado, o administrador usufrui de algumas permissões adicionais. Em sua tabela de configuração de usuário, estão registradas as informações de todos os administradores e funcionários do mercadinho. O administrador possui o poder de modificar as informações e excluir o perfil de qualquer usuário, proporcionando um controle abrangente sobre o gerenciamento de contas. Ademais, o administrador é responsável por adicionar novos usuários à empresa, com o botão correspondente situado no canto superior direito da tabela. Essa capacidade de gestão de usuários reforça a autonomia administrativa, permitindo a rápida incorporação de novos membros à equipe e garantindo a integridade das informações no sistema. Essa abordagem estruturada e hierárquica facilita não apenas a manutenção das contas existentes, mas também a expansão organizada do quadro de funcionários conforme as necessidades do Mercado Sunny.

Figura 24 – Tabela de configurações de usuário para o administrador.

Usuário











Logout

Configurações de Usuário

Adicionar Usuário

Show
10
entries

Search:

Registro	Nome	Permissões	Ações
123456	Ana Iara	Funcionário	 
333333	Isabely Graça	Funcionário	 
485329	Miguel Brito	Administrador	 
712846	Edna Sanchez	Administrador	 
987654	José Brito	Funcionário	 

Showing 1 to 5 of 5 entries

Previous1Next











Fonte: Autoria Própria (2023).

Figura 25 – Modal para editar usuário para o administrador.

The screenshot shows a web application interface for user management. A modal titled "Editar Usuário" is open, allowing an administrator to edit a user. The modal contains the following fields:

- Registro:** A text input field containing the value "123456".
- Nome:** A text input field containing the value "Ana lara".
- Senha:** A text input field containing the value "Nova senha".
- Permissões:** A dropdown menu currently showing "Funcionário".
- Salvar Alterações:** A blue button to save the changes.

The background interface includes a sidebar with "Usuário" and "Configurações de Usuário", a top bar with "Miguel Brito" and a "Logout" button, and a table of users with columns for "Registro", "Nome", "Permissões", and "Ações".

Registro	Nome	Permissões	Ações
123456	Ana lara	Funcionário	 
333333	Isabely Graça	Funcionário	 
485329	Miguel Brito	Administrador	 
712846	Edna Sanchez	Administrador	 
987654	José Brito	Funcionário	 











Fonte: Autoria Própria (2023).

Figura 26 – Modal para remover usuário para o administrador.

The screenshot shows the same web application interface, but with a modal titled "Remover Usuário" open. The modal contains the following elements:

- Deseja remover o usuário?:** A confirmation message.
- Não:** A grey button to cancel the removal.
- Sim:** A blue button to confirm the removal.











The background interface is identical to Figure 25, showing the user management sidebar, top bar, and table of users.

Registro	Nome	Permissões	Ações
123456	Ana lara	Funcionário	 
333333	Isabely Graça	Funcionário	 
485329	Miguel Brito	Administrador	 
712846	Edna Sanchez	Administrador	 
987654	José Brito	Funcionário	 

Fonte: Autoria Própria (2023).

Figura 27 – Modal para adicionar usuário.


The image shows a web application interface for user management. A modal titled "Adicionar Usuário" is open in the center. The modal contains the following fields: "Registro:" (text input), "Nome:" (text input), "Senha:" (password input), and "Permissão:" (dropdown menu with "Administrador" selected). At the bottom of the modal are two buttons: "Cancelar" and "Adicionar". In the background, the "Usuário" management page is visible, featuring a table with columns "Registro", "Nome", "Permissão", and "Ações". The table lists five users, with the first one being Miguel Brito. A "Logout" button is in the top right corner, and a "Adicionar Usuário" button is in the top right of the modal area.

Registro	Nome	Permissão	Ações
123456	Miguel Brito	Administrador	 
333333			 
485329			 
712846	Edna Sanchez	Administrador	 
987654	José Brito	Funcionário	 

Fonte: Autoria Própria (2023).

Figura 27 – Modal para editar funcionário.

The image shows the same web application interface, but with the "Editar Usuário" modal open. The modal contains the following fields: "Nome:" (text input with "José Brito" entered) and "Senha:" (password input with masked characters). At the bottom of the modal is a "Salvar Alterações" button. In the background, the "Usuário" management page is visible, showing a table with columns "Registro", "Nome", "Permissão", and "Ações". The table lists one user: José Brito. A "Previous" and "Next" button are at the bottom of the table, and a "Showing 1 to 1 of 1 entries" message is at the bottom left.

Registro	Nome	Permissão	Ações
987654	José Brito	Funcionário	

Fonte: Autoria Própria (2023).

10. CONCLUSÃO

Em síntese, este documento consistiu no detalhamento do projeto de sistema de mercado, abordando desde as etapas de levantamento de requisitos até a fase de prototipação. O sistema de gestão de mercado apresentado visa melhorar a eficiência operacional e a experiência do cliente em mercearias. Ao proporcionar um controle de estoque eficiente e acesso facilitado aos colaboradores, busca-se otimizar as operações diárias desses estabelecimentos.

A automação aprimorada do gerenciamento de estoque, com a introdução de gráficos visuais e intuitivos, não apenas garante um controle mais preciso dos produtos, mas também oferece aos usuários a capacidade de tomar decisões mais informadas. Essa abordagem visa criar uma interação mais dinâmica e orientada por dados entre o sistema e seus usuários, contribuindo para uma gestão mais eficaz do negócio.

Em última análise, o objetivo desse trabalho é oferecer uma solução prática e funcional que atenda às necessidades do mercado, simplificando as tarefas diárias e proporcionando um ambiente mais eficiente para a gestão de mercearias.

REFERÊNCIAS BIBLIOGRÁFICAS

BEZERRA, Eduardo. **Princípios de análise e projeto de sistemas com UML**. 5. Ed. Rio de Janeiro: Elsevier, 2015.

GUEDES, Gilleanes TA. **UML 2-Uma abordagem prática**. Novatec Editora, 2018.

BOOCH, Grady. **UML: guia do usuário**. Rio de Janeiro: Elsevie, 2006.