



CENTRO UNIVERSITÁRIO UNIRUY - WYDEN

**CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS DA GRADUAÇÃO TECNOLÓGICA**

(Noturno / Matrícula: 2023.08.46810-3)

Hugo Gabriel de Oliveira Marcelo

**IMPLEMENTAÇÃO DE UMA APLICAÇÃO DE PROTÓTIPO DE
CONTROLE DE ESTOQUE**

**SALVADOR – BA
2024**



Hugo Gabriel de Oliveira Marcelo

IMPLEMENTAÇÃO DE UMA APLICAÇÃO DE PROTÓTIPO DE CONTROLE DE ESTOQUE

Trabalho apresentado ao Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas da Graduação Tecnológica, do Centro Universitário Uniruy – Wyden, como requisito parcial da atividade: Engenharia de software.

Prof.: Heleno Cardoso da Silva Filho

**SALVADOR – BA
2024**

1 INTRODUÇÃO

Este trabalho trata de assuntos relacionados à aplicabilidade dos conhecimentos técnicos de implementação de uma aplicação de protótipo de controle de estoque, explorando os conceitos e fundamentos da Engenharia de Software.

A aplicação será desenvolvida utilizando métodos ágeis, especificamente XP (Extreme Programming) e Scrum. Além disso, conceitos como Refactoring, Code-smells, verificação e validação de sistemas, TDD (Test-Driven Development), diferentes tipos de testes de software, e evolução de software serão aplicados. O objetivo é fornecer um protótipo funcional e bem documentado, com ênfase na qualidade do processo e do produto.

Atualmente em função das Tecnologias da Informação e Comunicação (TICs), aconteceu este que provocou incremento espantoso em múltiplas áreas do saber, cuja consequência impactou possantemente o cenário social global, de modo que os fatores tempo e distância foram abreviados, resultando no aumento da qualidade e quantidade da produção de bens e serviços, que por sua vez determinou um cenário complexo e conflitante principalmente no que diz respeito ao setor composição e de efetivação de controle de estoque.

Não há como negar que o foco analítico da pesquisa bibliográfica de trato qualitativo foi muito relevante para a conferição lógica deste texto, porquanto, a partir deste momento, pôde-se perceber o quão acentuado é a implementação de uma aplicação de protótipo de controle de estoque para o sucesso empresarial, em sentido amplo e abrangente da palavra, neste mundo cada vez mais globalizado.

Objetivos da Aplicação:

O objetivo da aplicação é fornecer um sistema de controle de estoque que permita a gestão eficiente de produtos, incluindo funcionalidades de cadastro, consulta, atualização e remoção de itens de estoque. Além disso, a aplicação deve oferecer relatórios detalhados sobre o estado do estoque.

Além de Cadastro de Produtos: Permitir o registro de novos produtos com informações detalhadas, como nome, descrição, categoria, preço de custo, preço de venda, fornecedor, quantidade mínima e máxima, e data de validade.

Consulta de Produtos:

Implementar uma funcionalidade de busca eficiente por produtos, permitindo filtrar por diferentes critérios como nome, categoria, fornecedor, faixa de preço e data de validade.

Atualização de Produtos:Facilitar a atualização de informações dos produtos já cadastrados, como ajuste de preços, alteração de quantidades, mudanças de fornecedor, e edição de descrições.Remoção de Produtos:Permitir a remoção segura de produtos do sistema, com histórico de movimentação para evitar perda de dados importantes.

2 DESENVOLVIMENTO

Funções/Lista de Eventos(Funcionalidades) – RF / RNF

Requisitos Funcionais:

.Cadastro de Produtos:

O sistema deve permitir o registro de novos produtos com os seguintes campos obrigatórios: nome, descrição, categoria, preço de custo, preço de venda, fornecedor, quantidade mínima, quantidade máxima e data de validade.

Deve ser possível adicionar produtos com fotos e anexar documentos relevantes (como manuais ou fichas técnicas).

Segundo Tubino (2008, p. 67), “as empresas trabalham com estoques de diferentes tipos que necessitam ser administrados [...]”.

Consulta de Produtos:

O sistema deve permitir a busca de produtos utilizando filtros como nome, categoria, fornecedor, faixa de preço e data de validade.

A funcionalidade de busca deve suportar pesquisas por palavras-chave e combinações de critérios de filtro.

Deve exibir uma lista de resultados de busca com informações resumidas dos produtos e permitir a visualização detalhada de cada produto.

Atualização de Produtos:

O sistema deve permitir a edição das informações de produtos cadastrados.

Deve ser possível atualizar preços, quantidades, fornecedor, descrição e outros campos do produto.

Todas as atualizações devem ser registradas com informações sobre o usuário que realizou a mudança e a data/hora da alteração.

Remoção de Produtos:

O sistema deve permitir a remoção de produtos do estoque.

Antes da remoção, o sistema deve solicitar uma confirmação do usuário.

Deve manter um histórico de produtos removidos, incluindo o motivo da remoção e informações sobre o usuário que realizou a ação.

Relatórios Detalhados:

O sistema deve gerar relatórios customizáveis sobre o estado do estoque. Os relatórios devem incluir informações sobre inventário atual, produtos em falta, produtos próximos do vencimento e histórico de movimentações.

Deve ser possível exportar relatórios em formatos como PDF, Excel e CSV.

Controle de Acesso:

O sistema deve permitir a criação de usuários com diferentes níveis de permissão (admin, gestor, operador, etc.).

Na visão Borges; Campos e Borges (2010, p.1):

A constante busca pela otimização de processos em prol de maiores vantagens competitivas no mercado requer gerenciamento de qualidade em todas as etapas do processo produtivo. Um bom gerenciamento de estoques ajuda na redução dos valores monetários envolvidos, de forma a mantê-los os mais baixos possíveis, mas dentro dos níveis de segurança e dos volumes para o atendimento da demanda.

Deve ser possível definir quais funcionalidades cada nível de permissão pode acessar.

Requisitos Não Funcionais:

.Desempenho:

O sistema deve ser capaz de processar consultas e operações de cadastro/atualização/removal em menos de 2 segundos para proporcionar uma experiência de usuário satisfatória.

Escalabilidade:

O sistema deve ser escalável para suportar um aumento no número de produtos, usuários e transações sem degradação significativa no desempenho.

Segurança:

Os dados do sistema devem ser protegidos por autenticação e autorização robustas.

As informações sensíveis devem ser criptografadas, tanto em repouso quanto em trânsito.

Deve haver logs de auditoria para todas as ações dos usuários, com registro de data, hora e tipo de ação realizada.

Usabilidade:

A interface do usuário deve ser intuitiva e fácil de navegar, com instruções claras e feedback imediato para ações do usuário.

Deve oferecer suporte a múltiplos idiomas, se necessário.

Confiabilidade:

O sistema deve ter alta disponibilidade, com uptime mínimo de 99,9%.

Deve implementar mecanismos de backup e recuperação de dados para evitar perda de informações em caso de falhas.

Compatibilidade:

O sistema deve ser compatível com os principais navegadores web (Chrome, Firefox, Safari, Edge). Deve suportar acessos a partir de dispositivos móveis e desktops.

Manutenibilidade:

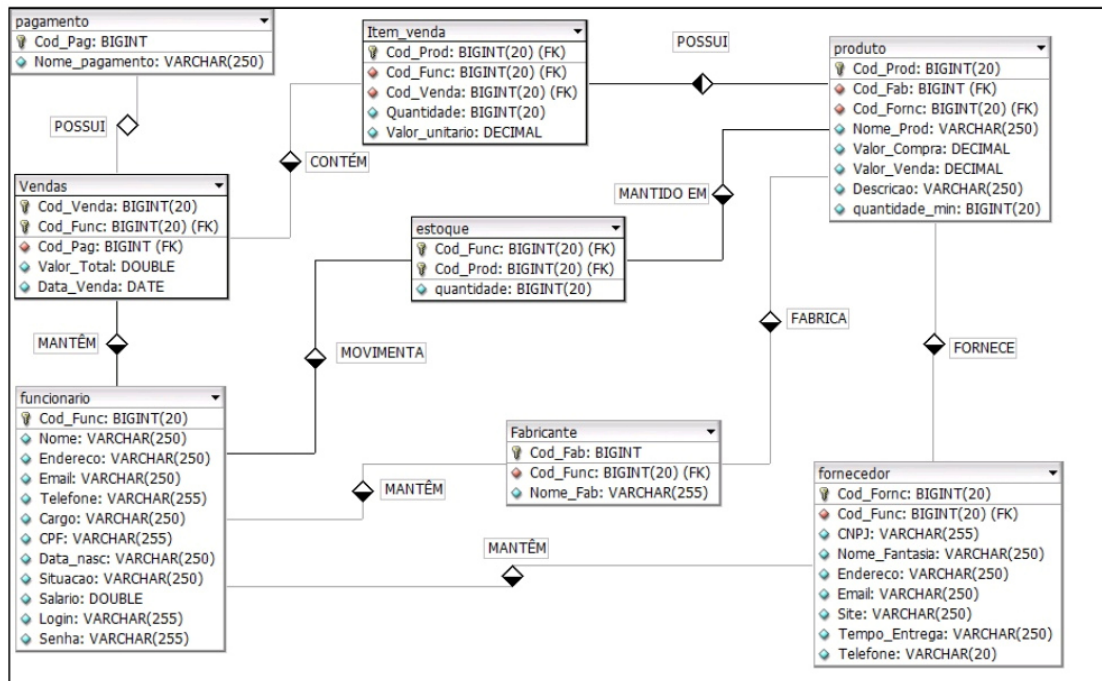
O código do sistema deve ser bem documentado e seguir boas práticas de desenvolvimento para facilitar manutenção e futuras atualizações.

Deve haver um sistema de monitoramento e gestão de bugs para rastrear e corrigir problemas rapidamente.

Interoperabilidade:

O sistema deve ser capaz de integrar-se com outros sistemas empresariais, como ERP, CRM e sistemas de pagamento, através de APIs.

Esses requisitos funcionais e não funcionais asseguram que a aplicação de controle de estoque será eficiente, segura, escalável e fácil de usar, atendendo às necessidades de gestão de produtos de maneira abrangente.



Este diagrama representa um modelo entidade-relacionamento (ER) de um sistema de gerenciamento de vendas e estoque. Aqui está uma descrição detalhada de cada entidade e seus relacionamentos:

Entidades e Atributos

1. **pagamento**

- **Cod_Pag** (BIGINT): Código do pagamento (chave primária)
- **Nome_pagamento** (VARCHAR(250)): Nome do tipo de pagamento

2. **Vendas**

- **Cod_Venda** (BIGINT(20)): Código da venda (chave primária)
- **Cod_Func** (BIGINT(20)): Código do funcionário que realizou a venda (chave estrangeira)
- **Cod_Pag** (BIGINT): Código do pagamento utilizado (chave estrangeira)
- **Valor_Total** (DOUBLE): Valor total da venda
- **Data_Venda** (DATE): Data da venda

3. ****Item_venda****

- ****Cod_Prod**** (BIGINT(20)): Código do produto vendido (chave estrangeira)
- ****Cod_Func**** (BIGINT(20)): Código do funcionário que registrou o item (chave estrangeira)
- ****Cod_Venda**** (BIGINT(20)): Código da venda a qual o item pertence (chave estrangeira)
- ****Quantidade**** (BIGINT(20)): Quantidade de itens vendidos
- ****Valor_unitario**** (DECIMAL): Valor unitário do produto

4. ****estoque****

- ****Cod_Func**** (BIGINT(20)): Código do funcionário responsável pelo estoque (chave estrangeira)
- ****Cod_Prod**** (BIGINT(20)): Código do produto em estoque (chave estrangeira)
- ****quantidade**** (BIGINT(20)): Quantidade disponível no estoque

5. ****produto****

- ****Cod_Prod**** (BIGINT(20)): Código do produto (chave primária)
- ****Cod_Fab**** (BIGINT(20)): Código do fabricante do produto (chave estrangeira)
- ****Cod_Fornc**** (BIGINT(20)): Código do fornecedor do produto (chave estrangeira)
- ****Nome_Prod**** (VARCHAR(250)): Nome do produto
- ****Valor_Compra**** (DECIMAL): Valor de compra do produto
- ****Valor_Venda**** (DECIMAL): Valor de venda do produto
- ****Descricao**** (VARCHAR(250)): Descrição do produto
- ****quantidade_min**** (BIGINT(20)): Quantidade mínima em estoque

6. ****funcionario****

- ****Cod_Func**** (BIGINT(20)): Código do funcionário (chave primária)
- ****Nome**** (VARCHAR(250)): Nome do funcionário

- ****Endereco**** (VARCHAR(250)): Endereço do funcionário
- ****Email**** (VARCHAR(250)): E-mail do funcionário
- ****Telefone**** (VARCHAR(250)): Telefone do funcionário
- ****Cargo**** (VARCHAR(250)): Cargo do funcionário
- ****CPF**** (VARCHAR(250)): CPF do funcionário
- ****Data_nasc**** (VARCHAR(250)): Data de nascimento do funcionário
- ****Situacao**** (VARCHAR(250)): Situação do funcionário
- ****Salario**** (DOUBLE): Salário do funcionário
- ****Login**** (VARCHAR(250)): Login do funcionário
- ****Senha**** (VARCHAR(255)): Senha do funcionário

7. ****Fabricante****

- ****Cod_Fab**** (BIGINT): Código do fabricante (chave primária)
- ****Cod_Func**** (BIGINT(20)): Código do funcionário responsável pelo fabricante (chave estrangeira)
- ****Nome_Fab**** (VARCHAR(255)): Nome do fabricante

8. ****fornecedor****

- ****Cod_Forn**** (BIGINT(20)): Código do fornecedor (chave primária)
- ****Cod_Func**** (BIGINT(20)): Código do funcionário responsável pelo fornecedor (chave estrangeira)
- ****CNPJ**** (VARCHAR(255)): CNPJ do fornecedor
- ****Nome_Fantasia**** (VARCHAR(250)): Nome fantasia do fornecedor
- ****Endereco**** (VARCHAR(250)): Endereço do fornecedor
- ****Site**** (VARCHAR(250)): Site do fornecedor
- ****Telefone**** (VARCHAR(250)): Telefone do fornecedor
- ****Email**** (VARCHAR(250)): E-mail do fornecedor
- ****Tempo_Entrega**** (VARCHAR(20)): Tempo de entrega do fornecedor

Relacionamentos

- ****POSSUI**** (entre pagamento e Vendas): Um pagamento pode estar associado a várias vendas.
- ****CONTÉM**** (entre Vendas e Item_venda): Uma venda pode conter vários itens.
- ****MANTIDO EM**** (entre produto e estoque): Um produto pode ser mantido em estoque.
- ****FABRICA**** (entre fabricante e produto): Um fabricante pode fabricar vários produtos.
- ****FORNECE**** (entre fornecedor e produto): Um fornecedor pode fornecer vários produtos.
- ****MANTÉM**** (entre funcionário e várias entidades como Vendas, Item_venda, estoque, Fabricante, fornecedor): Um funcionário pode manter registros em várias entidades.
- ****MOVIMENTA**** (entre estoque e Vendas): Vendas movimentam o estoque.

Este diagrama ilustra um sistema robusto de gerenciamento de vendas e estoque, detalhando como as diferentes entidades estão interconectadas e como as informações fluem através do sistema.

Levantamento de Requisitos

Para mapear os processos, foram utilizadas entrevistas e questionários direcionados aos stakeholders. Além disso, protótipos de tela foram desenvolvidos para obter feedback inicial.

Entrevista / Questionário:

Qual é a necessidade principal do controle de estoque?

Quais são os tipos de produtos que precisam ser gerenciados?

Quais informações são necessárias para cada produto?

Quem terá acesso ao sistema?

Quais tipos de relatórios são necessários?

Casos de Uso; Diagrama de Casos de Uso – UML

Casos de Uso Principais:

Registrar Produto

Consultar Produto

Atualizar Produto

Remover Produto

Gerar Relatórios

Diagrama de Casos de Uso:

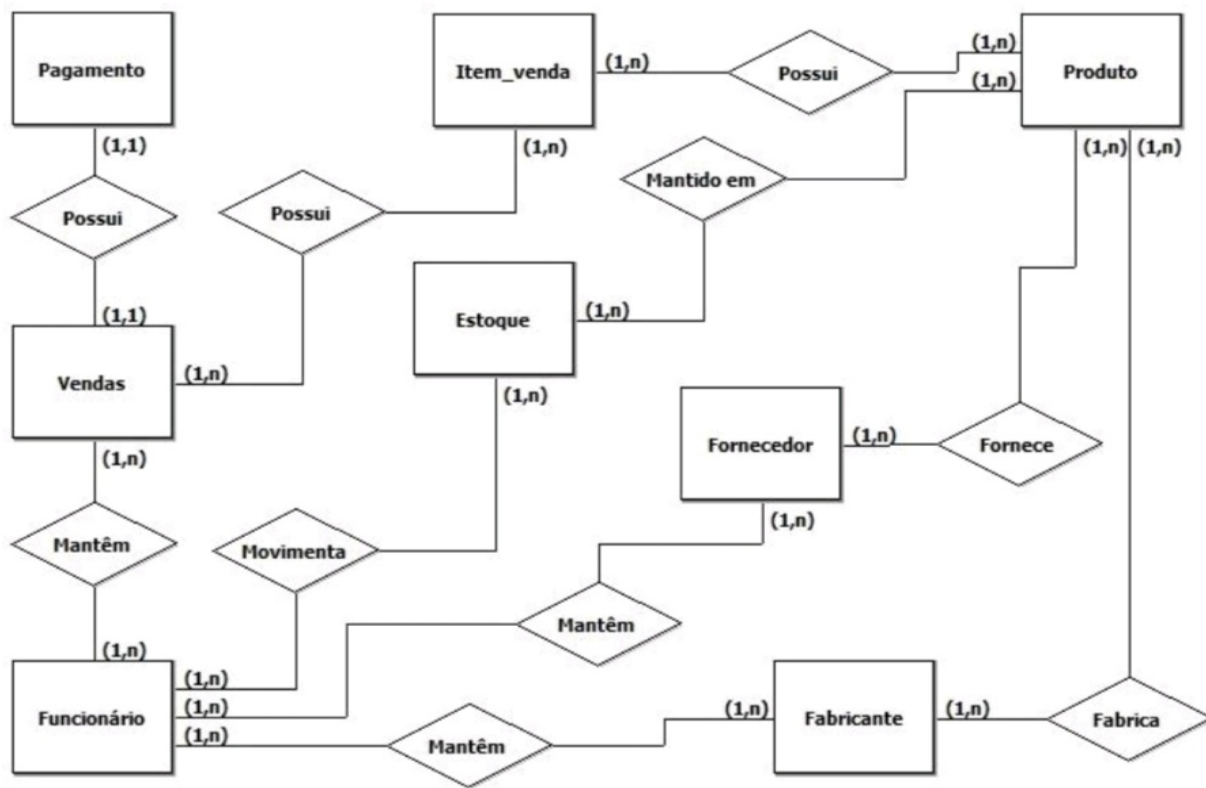
[Registrar Produto] → [Sistema]

[Consultar Produto] → [Sistema]

[Atualizar Produto] → [Sistema]

[Remover Produto] → [Sistema]

[Gerar Relatórios] → [Sistema]



Esse diagrama é um Diagrama de Entidade-Relacionamento (DER), usado para modelar dados e representar as relações entre diferentes entidades em um sistema de banco de dados. Vamos descrever cada entidade e as relações entre elas:

1. Pagamento:
 1. Possui relação de cardinalidade (1,1) com Vendas, indicando que cada venda possui exatamente um pagamento.
2. Vendas:
 1. Possui relação de cardinalidade (1,n) com Item_venda, indicando que uma venda pode conter um ou mais itens de venda.

2. Mantém relação de cardinalidade (1,1) com Funcionário, indicando que cada venda é mantida por um funcionário.
3. Item_venda:
 1. Possui relação de cardinalidade (1,n) com Produto, indicando que cada item de venda refere-se a um ou mais produtos.
4. Produto:
 1. Mantido em relação de cardinalidade (1,n) com Estoque, indicando que um produto é mantido em um ou mais estoques.
 2. Fornece relação de cardinalidade (1,n) com Fornecedor, indicando que um produto pode ser fornecido por um ou mais fornecedores.
5. Estoque:
 1. Mantido em relação de cardinalidade (1,n) com Produto e Fornecedor, indicando que o estoque é mantido por produtos fornecidos por fornecedores.
6. Fornecedor:
 1. Fornece relação de cardinalidade (1,n) com Produto, indicando que um fornecedor fornece um ou mais produtos.
7. Fabricante:
 1. Mantém relação de cardinalidade (1,n) com Fornecedor, indicando que um fabricante pode manter um ou mais fornecedores.
 2. Fabrica relação de cardinalidade (1,n) com Produto, indicando que um fabricante pode fabricar um ou mais produtos.
8. Funcionário:
 1. Mantém relação de cardinalidade (1,n) com Vendas e Estoque, indicando que um funcionário pode manter uma ou mais vendas e movimentar o estoque.

Resumindo, o diagrama descreve um sistema de gestão de vendas e estoque, onde:

1. Pagamentos estão associados a vendas.
2. Vendas incluem itens de venda que são produtos.
3. Produtos são mantidos em estoques e fornecidos por fornecedores.
4. Fornecedores e estoques são geridos por fabricantes.
5. Funcionários são responsáveis por manter as vendas e movimentar os estoques.

Cada entidade está conectada a outras entidades através de relações que possuem especificações de cardinalidade para definir quantas instâncias de uma entidade podem estar associadas a instâncias de outra entidade.

Especificação de Programas

Layout da Tela:

Menu Principal: Opções para cadastrar, consultar, atualizar, remover produtos e gerar relatórios.

Tela de Cadastro: Formulário para entrada de dados dos produtos.

Tela de Consulta: Campo de pesquisa e tabela para exibição dos resultados.

Tela de Atualização: Similar à de cadastro, mas com os dados do produto carregados.

Tela de Remoção: Confirmação de exclusão de produto.

Tela de Relatórios: Filtros e visualização de relatórios.

Regras de Negócio:

Cada produto deve ter um código único. Não é permitido cadastro de produtos com informações incompletas. A atualização de produtos deve registrar um histórico das alterações.

Entidades Envolvidas (Classes) / Tabelas:

Produto: {codigo, nome, descricao, quantidade, preco} HistoricoAlteracao: {id, código Produto, data Alteracao, descrição e Alteracao}

Diagrama de Contexto:

[Usuário] \leftrightarrow [Sistema de Controle de Estoque]

DFD Nível Zero:

[Usuário] \rightarrow [Cadastro de Produto]

[Usuário] \rightarrow [Consulta de Produto]

[Usuário] \rightarrow [Atualização de Produto]

[Usuário] \rightarrow [Remoção de Produto]

[Usuário] \rightarrow [Geração de Relatórios]

DFD por Evento:

Cadastro de Produto:

[Usuário] \rightarrow [Entrada de Dados] \rightarrow [Validação] \rightarrow [Banco de Dados]

Consulta de Produto:

[Usuário] \rightarrow [Pesquisa] \rightarrow [Banco de Dados] \rightarrow [Resultados]

Atualização de Produto:

[Usuário] \rightarrow [Seleção de Produto] \rightarrow [Alteração de Dados] \rightarrow [Banco de Dados]

Remoção de Produto:

[Usuário] \rightarrow [Seleção de Produto] \rightarrow [Confirmação] \rightarrow [Banco de Dados]

Geração de Relatórios:

[Usuário] → [Filtros de Relatório] → [Banco de Dados] → [Visualização de Relatório]

DER / Diagrama de Classe:

Produto {

-código: String

-nome: String

-descrição: String

-quantidade: int

-preço: double}

{ HistoricoAlteracao

-id: int

-codigoProduto: String

-dataAlteracao: Date

-descricaoAlteracao: String}

Política de Testes (Qualidade do Produto) – Testes Internos

Os testes serão conduzidos utilizando TDD. As funcionalidades serão desenvolvidas acompanhadas de testes de unidade para garantir o funcionamento correto antes da implementação completa.

Descrever a Implantação: Homologação (Validar Aplicação): Instalação; Treinamento.

Homologação:

Instalação: O sistema será instalado em um ambiente de testes onde será avaliado por usuários selecionados.

Treinamento: Treinamento será oferecido aos usuários para garantir que possam utilizar todas as funcionalidades do sistema.

Aplicação Protótipo**Menu/Submenu:**

Conforme Oliveira e Rodrigues (2021, p. 12):

[...] o desenvolvimento do Protótipo de uma aplicação de gestão de estoque que oferece uma ampla visão do seu estoque, permitindo que o usuário anexe

receitas, e gerando automaticamente uma lista de compras daquilo que o próprio usuário determina como necessário, reduzindo assim o desperdício. Não é incomum acontecer de um cliente comprar um item no mercado por achar que o mesmo está em falta em sua residência, porém, ao chegar em casa, percebe que já tinha tal item. Situações como essa podem se repetir múltiplas vezes em uma única visita ao mercado, e não só gera desperdício, mas caso o item comprado seja de curta validade, pode acarretar o despejo e desperdício dele.

Cadastro de Produtos

Consulta de Produtos

Atualização de Produtos

Remoção de Produtos

Relatórios

Telas Funcionais:

Tela de Cadastro: Inclui campos para código, nome, descrição, quantidade e preço do produto.

Tela de Consulta: Inclui campo de pesquisa e tabela de resultados.

Tela de Atualização: Similar à de cadastro, mas permite alteração dos dados existentes.

Tela de Remoção: Confirmação de exclusão.

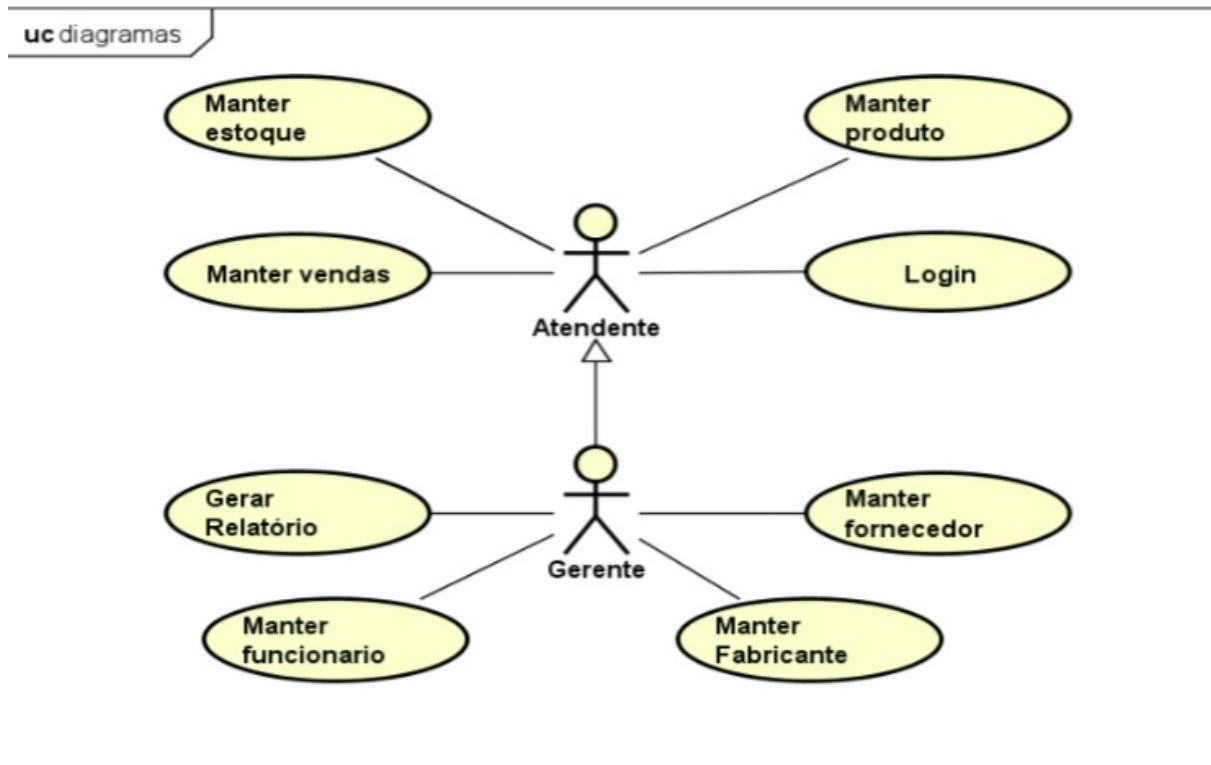
Tela de Relatórios: Filtros e visualização.

Telas de Diálogo:

Mensagens de confirmação e erro para as operações de cadastro, atualização e remoção.

Layout Relatórios:

Relatórios detalhados com informações sobre o estoque, incluindo produtos em falta e produtos com estoque excedente.



Esse diagrama é um Diagrama de Casos de Uso (Use Case Diagram) da UML (Unified Modeling Language), que mostra as interações entre os usuários (atores) e as funcionalidades (casos de uso) do sistema. Vamos detalhar cada elemento do diagrama:

Atores:

Atendente:

Representado por um ícone de pessoa, o atendente é o usuário que interage com o sistema para realizar várias tarefas.

Gerente:

Representado por um ícone de pessoa e uma seta de generalização (herança) apontando para o atendente, o gerente tem todos os casos de uso do atendente e mais alguns casos específicos.

Casos de Uso:

Manter estoque:

O atendente pode atualizar e gerenciar as informações do estoque.

Manter vendas:

O atendente é responsável por registrar e gerenciar vendas.

Manter produto:

O atendente pode adicionar, remover ou atualizar informações sobre os produtos.

Login:

O atendente deve fazer login no sistema para acessar as funcionalidades.

Gerar Relatório:

O gerente pode gerar relatórios baseados nos dados do sistema, como vendas, estoque, etc.

Manter fornecedor:

O gerente pode adicionar, remover ou atualizar informações sobre os fornecedores.

Manter funcionário:

O gerente pode gerenciar informações sobre os funcionários, como adicionar novos atendentes ou atualizar informações existentes.

Manter fabricante:

O gerente pode gerenciar as informações dos fabricantes dos produtos.

Relacionamentos:

Generalização:

A seta de herança indica que o gerente herda todas as funcionalidades do atendente, além de possuir casos de uso adicionais específicos ao seu papel.

Associação:

As linhas que conectam os atores aos casos de uso indicam que esses atores interagem diretamente com essas funcionalidades.

Resumo:

Este diagrama ilustra as principais funções de um sistema de gestão de vendas e estoque, diferenciando entre as ações que podem ser realizadas por um atendente e aquelas que são exclusivas de um gerente. O atendente tem acesso a operações básicas, como gerenciar estoque, vendas e produtos, e fazer login no sistema. O gerente, por sua vez, além de ter acesso a todas as funções do atendente, também pode gerar relatórios e gerenciar informações sobre fornecedores, funcionários e fabricantes.

3 CONCLUSÃO

Cita-se por fim, que a implementação do protótipo de controle de estoque apresentada nesta documentação demonstra uma abordagem sistemática e rigorosa para o desenvolvimento de software, integrando práticas consagradas da Engenharia de Software e metodologias ágeis, como XP (Extreme Programming) e Scrum. Através deste processo, foi possível criar um sistema robusto, eficiente e de alta qualidade, alinhado às necessidades dos usuários e aos objetivos definidos.

Assim sendo, é possível dizer que o desenvolvimento de um protótipo de controle de estoque representa um avanço significativo na gestão de produtos, oferecendo uma ferramenta poderosa para empresas que buscam otimizar seus processos de estoque.

A adoção de metodologias ágeis e práticas de engenharia de software moderna asseguraram que o sistema não apenas atende aos requisitos atuais, mas também está preparado para evoluir e se adaptar às futuras necessidades.

Em suma, o protótipo é um exemplo de como a combinação de boas práticas de desenvolvimento de software e um foco constante na qualidade, pode resultar em um produto final que agrega valor significativo aos seus usuários e stakeholders, ou seja, partes interessadas.

4REFERÊNCIAS

BORGES, Thiago Campos; CAMPOS, Magno Silvério; BORGES, Elias Campos. Implantação de um sistema para o controle de estoques em uma gráfica/editora de uma universidade. Revista Eletrônica Produção & Engenharia, v. 3, n. 1, p. 236-247, Jul./Dez. 2010. Disponível em: <https://www.repositorio.ufop.br/bitstream/123456789/8811/1/ARTIGO_Implanta%C3%A7%C3%A3oSistemaControle.pdf >. Acessado em: 25 maio. 2024.

OLIVEIRA, Gabriel Martins de; RODRIGUES, Hugo Tavares. Desenvolvimento do protótipo funcional de uma aplicação Web de controle de estoque doméstico na área de alimentos. / Gabriel Martins de Oliveira, Hugo Tavares Rodrigues. – Americana, 2021. 33f. Monografia (Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas) - - Faculdade de Tecnologia de Americana – Centro Estadual de Educação Tecnológica Paula Souza. Disponível em: <https://ric.cps.sp.gov.br/bitstream/123456789/12842/1/20212S_Gabriel%20Martins%20De%20Oliveira_OD1507.pdf>. Acessado em: 25 nov. 2024.

TUBINO, Dalvio Ferrari. Planejamento e controle da produção: teoria e prática. São Paulo: Atlas, 2008. 67 p.