



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

JOÃO VITOR VARELA LANDIM BARRETO

SISTEMA DE GERENCIAMENTO DE PESCA

**Assis/SP
2024**



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

JOÃO VITOR VARELA LANDIM BARRETO

SISTEMA DE GERENCIAMENTO DE PESCA

Projeto de desenvolvimento apresentado ao curso de Análise e Desenvolvimento de Sistemas do Instituto Municipal de Ensino Superior de Assis – IMESA e a Fundação Educacional do Município de Assis – FEMA, como requisito parcial à obtenção do Certificado de Conclusão.

Orientando(a): João Vitor Varela Landim Barreto

Orientador(a): Dr. Alex Sandro Romeo de Souza Poletto

**Assis/SP
2024**

Barreto, João Vitor Varela Landim

B273s Sistema de gerenciamento de pesca / João Vitor Varela Landim Barreto. -- Assis, 2024. -- 29p.

Trabalho de Conclusão de Curso (Graduação em Análise e Desenvolvimento de Sistemas) -- Fundação Educacional do Município de Assis (FEMA), Instituto Municipal de Ensino Superior de Assis (IMESA), 2024.

Orientador: Prof. Dr. Alex Sandro Romeo de Souza Poletto.

1. Controle de estoque. 2. Sistemas de gestão empresarial. 3. Sistemas de informação gerencial. I Poletto, Alex Sandro Romeo. II Título.

CDD 003

AGRADECIMENTOS

Em primeiro lugar, agradeço a Deus por me permitir a possibilidade de estar aqui, me fornecer dedicação, e força para chegar onde eu sempre sonhei. Agradeço, à minha mãe, Iara, à minha avó, Laura, ao meu pai, Douglas, e a minha namorada, Marcella, por toda a ajuda e apoio que me forneceram até aqui.

Também agradeço ao meu orientador e professor, Alex Sandro Romeo de Souza Poletto, por toda a ajuda e paciência que me prestou ao decorrer do ano. Seu conhecimento e comprometimento foram essenciais para a execução do projeto.

E por último, mas não menos importante, agradeço a todos os meu amigos de sala, por toda a nossa trajetória até aqui, e por todos os conhecimentos que foram trocados.

RESUMO

O projeto apresenta o desenvolvimento de um sistema para o gerenciamento de uma loja de pesca, com o propósito de otimizar o controle de estoque e caixa, facilitando a administração eficiente dos produtos cadastrados, com funcionalidades de cadastro, consulta, e atualização dos produtos e clientes. A construção do sistema utiliza as tecnologias Java e MySQL, com a finalidade de trazer redução de erros, aumento de produtividade e contribuir para uma melhor gestão da loja.

Palavras-chave: Controle de estoque, Cadastro de produtos, Redução de erros.

ABSTRACT

The project presents the development of a system for managing a grocery store. fishing, with the purpose of optimizing stock and cash control, facilitating administration of registered efficient products, with registration, consultation and update features products and customers. The construction of the system uses Java and MySQL technologies, with the specifically to reduce errors, increase productivity and contribute to better store management.

Keywords: Inventory control, Product registration, Error reduction.

LISTA DE ILUSTRAÇÕES

Figura 1: Diagrama de Entidade e Relacionamento - Sistema	13
Figura 2: Diagrama de Classes – Sistema	14
Figura 3: Diagrama de Casos de Uso – Administrador e Funcionário	15
Figura 4: Criação de cliente	16
Figura 5: Tela de cadastro de cliente	17
Figura 6: Criação de funcionário	18
Figura 7: Tela de cadastro de funcionário	19

LISTA DE ABREVIATURAS E SIGLAS

ERP	Enterprise Resource Planning
UML	Unified Modeling Language
DER	Diagrama Entidade Relacionamento
JUDE	Java and UML Developers Environment
SQL	Structured Query Language
IDE	Integrated Development Environment
STMT	Statement

SUMÁRIO

1.	INTRODUÇÃO	4
1.1.	OBJETIVOS	4
1.1.1.	OBJETIVO GERAL	4
1.1.2.	OBJETIVO ESPECÍFICO.....	5
1.2.	PÚBLICO ALVO	5
1.3.	JUSTIFICATIVA	5
1.5.	MÉTODO DE DESENVOLVIMENTO	6
2.	TECNOLOGIAS	8
2.1.	UML	8
2.1.1.	DIAGRAMA DE CLASSE	8
2.1.2.	DIAGRAMA DE CASOS DE USO	8
2.1.3.	DIAGRAMA ENTIDADE RELACIONAMENTO.....	9
2.1.4.	ASTAH UML.....	9
2.1.5.	JAVA.....	9
2.1.6.	JAVA SWING	10
3.	REQUISITOS.....	11
3.1.	DESCRIÇÃO DOS REQUISITOS	11
3.1.1.	MANTER CLIENTE.....	11
3.1.2.	MANTER FUNCIONÁRIO	11
3.1.3.	MANTER FORNECEDOR.....	11
3.1.4.	MANTER PRODUTO	12
3.1.5.	EFETUAR VENDAS	12
3.1.6.	EMITIR RELATÓRIOS DE VENDAS MENSAIS	12
4.	DIAGRAMAS.....	13
4.1.	DIAGRAMA E.R. (ENTIDADE RELACIONAMENTO)	13
4.2.	DIAGRAMA DE CLASSES.....	14
4.3.	DIAGRAMA DE CASOS DE USO	15
5.	DESENVOLVIMENTO DO TRABALHO	16

5.1. ESTRUTURA DO BACKEND	16
5.1.1. CRIAÇÃO DO CLIENTE	16
5.1.2. TELA PARA CADASTRO DE CLIENTE	17
5.1.3. CRIAÇÃO DO FUNCIONÁRIO	18
5.1.4. TELA PARA CADASTRO DE FUNCIONÁRIO	19
6. CONCLUSÃO.....	20
6.1. TRABALHOS FUTUROS	20
REFERÊNCIAS	21

1. INTRODUÇÃO

A gestão eficiente de uma loja de pesca requer soluções específicas para superar os desafios do mercado. Nesse contexto, o software surge como uma ferramenta essencial, oferecendo recursos integrados e personalizados. O sistema possibilita o controle do estoque, gerenciamento de vendas, e controle financeiro.

Este projeto desenvolveu um software exclusivo para o gestor de uma loja de pesca. O sistema permite o controle do estoque, emissão de relatórios personalizados e monitoramento do fluxo de caixa, fornecendo informações essenciais para uma gestão eficiente e tomada de decisões estratégicas.

De acordo com Gestão Click (2023), lojas de pesca são espaços com relativo grau de complexidade, e para facilitar a vida dos empresários, o sistema contará com recursos modernos e práticos como o controle financeiro, controle de estoque, vendas, e perfis de acesso.

Segundo Santiago et al. (2018), as empresas têm a necessidade de controlar diversas funções como vendas, estoque, movimento bancário, dentre outras. O controle dessas funções pode ser realizado de forma manual, gerando um trabalho cansativo e sujeito a erros, ou pode ser feito com mais eficiência e precisão com o auxílio da tecnologia da informação, e a adoção de um sistema próprio para a empresa.

De acordo com Oliveira (2017), para que exista apoio nas atividades, independente do ramo que esteja seguindo, a utilização de sistemas não pode ser dispensada. Desta forma as organizações seguem modernizando seus softwares acompanhando a evolução tecnológica, e com isso existe uma grande perspectiva de crescimento para o mercado de um tipo de sistema de informação conhecido como *Enterprise Resource Planning (ERP)*.

1.1. OBJETIVOS

1.1.1. OBJETIVO GERAL

Este trabalho de conclusão de curso tem por objetivo geral o desenvolvimento de um software para uma loja de artigos de pesca onde o usuário terá acesso as vendas, estoque e caixa.

1.1.2. OBJETIVO ESPECÍFICO

Como objetivo específico, foi desenvolvido um sistema interno para facilitar o gerenciamento das vendas, entrada e saída do estoque e a movimentação do caixa. Pretende-se melhorar a precisão dos registros de vendas, ter um controle de estoque assertivo, e também um monitoramento financeiro mais eficiente.

- Cadastros: o funcionário será capaz de criar o cadastro de clientes, produtos, funcionários e fornecedores.
- Relatórios em geral: o funcionário tem acesso ao relatório de vendas, compras e clientes.
- Utilização de um sistema desktop, ao qual o usuário tem acesso por download do programa na máquina.

1.2. PÚBLICO ALVO

O sistema de gerenciamento de pesca foi desenvolvido para uma empresa na cidade de Assis-SP, com a finalidade de atualizar e melhorar a gestão do comércio e suas tarefas.

Os usuários que vão utilizar o software são profissionais de vendas, que prestam atendimento ao cliente, buscam ter eficiência no serviço e um bom fluxo de caixa, pois adquirindo o sistema a empresa procura obter um controle financeiro assertivo, e uma base de clientes e dados.

1.3. JUSTIFICATIVA

A gestão de uma loja é muito importante, ainda mais quando ela conta com um sistema próprio para fazer todo o processo de controle. Um software apropriado pode fornecer muitas vantagens como: registro de vendas: um registro digital pode ajudar com o histórico de transações e permite identificar o padrão de compra. Com isso o gestor poderá criar promoções e planos de vendas. Controle de caixa: O sistema possibilita que o usuário tenha acesso a entrada e saída de valores no caixa. Isso é importante para que as finanças da empresa fiquem em dia e equilibradas no fim do mês, diminuindo o risco de perda de dinheiro. Controle de estoque: Ter uma estocagem de artigos de pesca é indispensável para atender todas as demandas dos clientes, e não correr o risco de perdas financeiras por conta da falta de produtos no estoque. A gestão do inventário permite ter o controle e identificar itens com falta no estoque e assegurar que esteja sempre atualizado.

1.4. ESTRUTURA DO TRABALHO

O presente trabalho está organizado em cinco capítulos:

- **Capítulo 1 - Introdução:** apresentando os objetivos e as justificativas para a execução do mesmo.
- **Capítulo 2 - Tecnologias:** destaca as tecnologias utilizadas no projeto.
- **Capítulo 3 - Requisitos:** aborda os requisitos do sistema.
- **Capítulo 4 - Diagramas:** mostra os diagramas do projeto.
- **Capítulo 5 - Desenvolvimento do trabalho:** relata o desenvolvimento do projeto, com imagens do código e do programa.
- **Capítulo 6 - Conclusão e trabalhos futuros:** desenvolve a conclusão do trabalho, e projetos futuros.
- **Capítulo 7 - Cronograma:** cronograma do projeto, ao decorrer do ano.

1.5. MÉTODO DE DESENVOLVIMENTO

Para atingir os objetivos aqui estabelecidos, elaborou-se um esquema metodológico que está organizado em 5 fases:

Na fase 1, consiste em uma reunião com o cliente para entender e realizar um levantamento de quais serão os requisitos essenciais para a necessidade do usuário, e buscando atender as demandas propostas.

O protótipo funcional foi desenvolvido na fase 2, seguindo o levantamento dos requisitos, e a apresentação para o cliente avaliar.

Seguindo para a fase 3 no desenvolvimento do projeto do software, foi utilizada a metodologia *UML* para apresentar o trabalho. Foram produzidos os seguintes artefatos *UML*: diagrama de caso de uso, diagrama de classe e o diagrama de entidade relacional. A plataforma Astah *UML*, por meio da utilização para a criação dos diagramas de classe e caso de uso, já o software DB Designer foi manuseado para a geração do diagrama de entidade relacional. O banco de dados utilizado no sistema, foi o MySQL, pois é um banco de fácil entendimento, conta com bons recursos de segurança e proporciona boa velocidade e desempenho. A linguagem de programação utilizada foi o Java, junto a biblioteca Java Swing.

O teste do software ocorre na fase 4, onde é realizada a verificação da integridade do sistema e identificadas as possíveis falhas.

Na fase 5, foi feita a escrita final do TCC, que concluiu o projeto e finalizou o sistema.

2. TECNOLOGIAS

2.1. UML

De acordo com IBM (2021), Os diagramas *UML* ilustram os aspectos qualificáveis de um sistema que podem ser descritos visualmente, como relacionamentos, comportamento, estrutura e funcionalidade. Por exemplo, um diagrama de classe descreve a estrutura do sistema ou os detalhes de uma implementação, enquanto um diagrama de sequência mostra a interação entre objetos com o tempo. Em um diagrama *UML*, os elementos de diagrama representam visualmente os classificadores em um sistema ou aplicativo. Esses classificadores são representações em diagrama de um elemento de origem. Diagramas *UML* fornecem visualizações de elementos de origem; no entanto, elementos de diagramas não têm valor semântico.

2.1.1. DIAGRAMA DE CLASSE

Segundo IBM (2021), Em *UML*, diagramas de classes são um dos seis tipos de diagramas estruturais. Os diagramas de classe são fundamentais para o processo de modelagem de objetos e modelam a estrutura estática de um sistema. Dependendo da complexidade de um sistema, é possível utilizar um único diagrama de classe para modelar um sistema inteiro ou vários diagramas de classe para modelar os componentes de um sistema.

Os diagramas de classe são as cópias do sistema ou subsistema. Você pode utilizar os diagramas de classe para modelar os objetos que compõem o sistema, para exibir os relacionamentos entre os objetos e para descrever o que esses objetos fazem e os serviços que eles fornecem.

2.1.2. DIAGRAMA DE CASOS DE USO

Conforme IBM (2021), os diagramas de caso de uso modelam o comportamento de um sistema e ajudam a capturar os requisitos do sistema. Os diagramas de caso de uso descrevem funções de alto nível e escopo de um sistema. Esses diagramas também identificam as interações entre o sistema e seus agentes. Os casos de uso e os agentes nos diagramas de caso de uso descrevem o que o sistema faz e como os agentes o usam, mas não como o sistema opera internamente.

Os diagramas de casos de uso ilustram e definem o contexto e os requisitos de um sistema inteiro ou das partes importantes dele. É possível modelar um sistema complexo com um

único diagrama de caso de uso ou ainda criar muitos diagramas de caso de uso para modelar os componentes do sistema. Normalmente, os diagramas de casos de uso são desenvolvidos nas fases iniciais de um projeto e são consultados em todo o processo de desenvolvimento.

2.1.3. DIAGRAMA ENTIDADE RELACIONAMENTO

De acordo com Awari (2023), Diagrama Entidade-Relacionamento (*DER*) é uma ferramenta utilizada na modelagem de dados. Ele permite visualizar de forma clara as relações entre as entidades de um sistema e como essas entidades se relacionam entre si.

Basicamente, um *DER* é um tipo de diagrama que mostra as entidades, os atributos e as relações entre elas. As entidades são objetos que existem no mundo real ou no mundo virtual, como pessoas, produtos, pedidos, entre outros. Os atributos são características das entidades, como nome, endereço, telefone, entre outros. Já as relações representam como as entidades estão conectadas entre si, como por exemplo, um cliente pode realizar vários pedidos, mas um pedido só pode pertencer a um cliente.

2.1.4. ASTAH UML

Segundo TechTudo (2022), Astah Community é um software para modelagem *UML* (Unified Modeling Language – Linguagem de Modelagem Unificada) com suporte a UML 2, desenvolvido pela Change Vision, Inc e disponível para sistemas operacionais Windows 64 bits. Anteriormente conhecido por *JUDE*, um acrônimo de Java and *UML* Developers Environment (Ambiente para Desenvolvedores *UML* e Java).

Astah Community disponibiliza para desenvolvimento, os diagramas de Classes, Casos de Uso, Sequência, Comunicação, Máquina de Estados, Atividade, Componentes, Implantação e Diagrama de Estrutura Composta.

2.1.5. JAVA

De acordo com Bessa (2023), O Java é uma das plataformas mais utilizadas para o desenvolvimento de aplicações ao redor do mundo. Além da plataforma, segundo dados da Pesquisa “Tecnologias mais populares de 2022”, realizada pelo StackOverflow, a linguagem Java figura como a 6ª mais utilizada por pessoas desenvolvedoras, correspondendo a 33% do total pesquisado. O Java, como plataforma de programação, nasceu no ano de 1995 dentro dos laboratórios da empresa Sun Microsystem como resultado de uma extensa pesquisa científica e tecnológica. A plataforma Java entrega um ambiente completo para o

2.1.6. JAVA SWING

Conforme Awari (2023), Java Swing é uma biblioteca de interface gráfica de usuário para o Java, que permite criar aplicativos com interfaces gráficas intuitivas para o usuário. Este oferece uma ampla variedade de componentes de interface gráfica de usuário, como botões, caixas de seleção, campos de texto, barras de rolagem, tabelas, etc., que permitem criar interfaces gráficas elegantes e profissionais. Além disso, possui um grande conjunto de recursos que permitem personalizar a aparência e o comportamento desses componentes para se adequar às necessidades do aplicativo.

A criação de interfaces gráficas em Java Swing é relativamente fácil e intuitiva, mesmo para desenvolvedores iniciantes. Além disso, ele é altamente compatível com outras bibliotecas Java, permitindo que os desenvolvedores adicionem recursos avançados de processamento de dados e lógica de negócios aos seus aplicativos de interface gráfica.

2.1.7. BANCO DE DADOS MYSQL

Segundo Cardoso (2023), O MySQL é um sistema de gerenciamento de banco de dados relacional de código aberto. O software funciona em um modelo de arquitetura, permitindo dessa forma que múltiplos clientes se conectem ao servidor e acessem e manipulem os dados armazenados nos bancos de dados. Os dados são organizados em tabelas compostas de linhas e colunas. A partir da linguagem de consulta estruturada (*SQL*), é possível realizar operações de criação, modificação e consulta de dados. A ferramenta fornece uma ampla gama de recursos e funcionalidades, incluindo suporte a transações, indexação eficiente, controle de acesso e segurança dos dados.

Além disso, o MySQL é conhecido por sua estabilidade, confiabilidade e facilidade de uso. Ele conta com uma grande comunidade de desenvolvedores e usuários em todo o mundo, o que contribui para seu contínuo aprimoramento e suporte.

3. REQUISITOS

Os requisitos obtidos para o Sistema de Gerenciamento de Pesca são:

- Manter cliente
- Manter funcionário
- Manter fornecedor
- Manter produto
- Manter administrador
- Efetuar vendas
- Emitir relatórios de vendas mensais

3.1. DESCRIÇÃO DOS REQUISITOS

3.1.1. MANTER CLIENTE

O requisito manter cliente oferece a gestão sobre o cliente, para cadastramento do mesmo com todas as informações necessárias, permitindo que o usuário tenha acesso para cadastrar, atualizar, consultar e excluir os dados, proporcionando um relacionamento eficiente com o consumidor, e que todas as suas informações estejam visíveis.

3.1.2. MANTER FUNCIONÁRIO

Manter funcionário abrange o controle dos dados dos empregados, como o cadastro, atualização, exclusão e visualização, garantindo que o gestor defina o nível de acesso e o cargo do colaborador.

3.1.3. MANTER FORNECEDOR

Em manter fornecedor, o sistema permite o gerenciamento total dos fornecedores, com a inclusão do cadastro de novos parceiros, atualização de informações e o histórico de fornecimentos, além de oferecer consultas e a exclusão de fornecedores inativos.

3.1.4. MANTER PRODUTO

Já no requisito manter produto, ele possibilita o controle sobre o cadastro dos produtos e gerenciamento dos itens no estoque, possibilitando a adição, modificação e exclusão de produtos, e também garantir que o software tenha processamento preciso das quantidades e características de cada um.

3.1.5. EFETUAR VENDAS

Em efetuar vendas, esse requisito cadastra a venda, guardando todas as informações necessárias como o ID da venda, nome do cliente, os itens que compõem, a data que foi registrada e o valor da venda.

3.1.6. EMITIR RELATÓRIOS DE VENDAS MENSAS

Por último, o requisito emitir relatório de vendas mensais tem o objetivo de gerar relatórios detalhados sobre o desempenho das vendas em um período específico, permitindo uma análise crucial do volume de faturação, itens mais vendidos e o padrão de compras dos clientes, auxiliando na tomada de decisões.

4. DIAGRAMAS

4.1. DIAGRAMA E.R. (ENTIDADE RELACIONAMENTO)

Conforme Awari (2023), o Diagrama Entidade-Relacionamento é uma ferramenta utilizada na modelagem de dados. Ele permite visualizar de forma clara as relações entre as entidades de um sistema e como essas entidades se relacionam entre si.

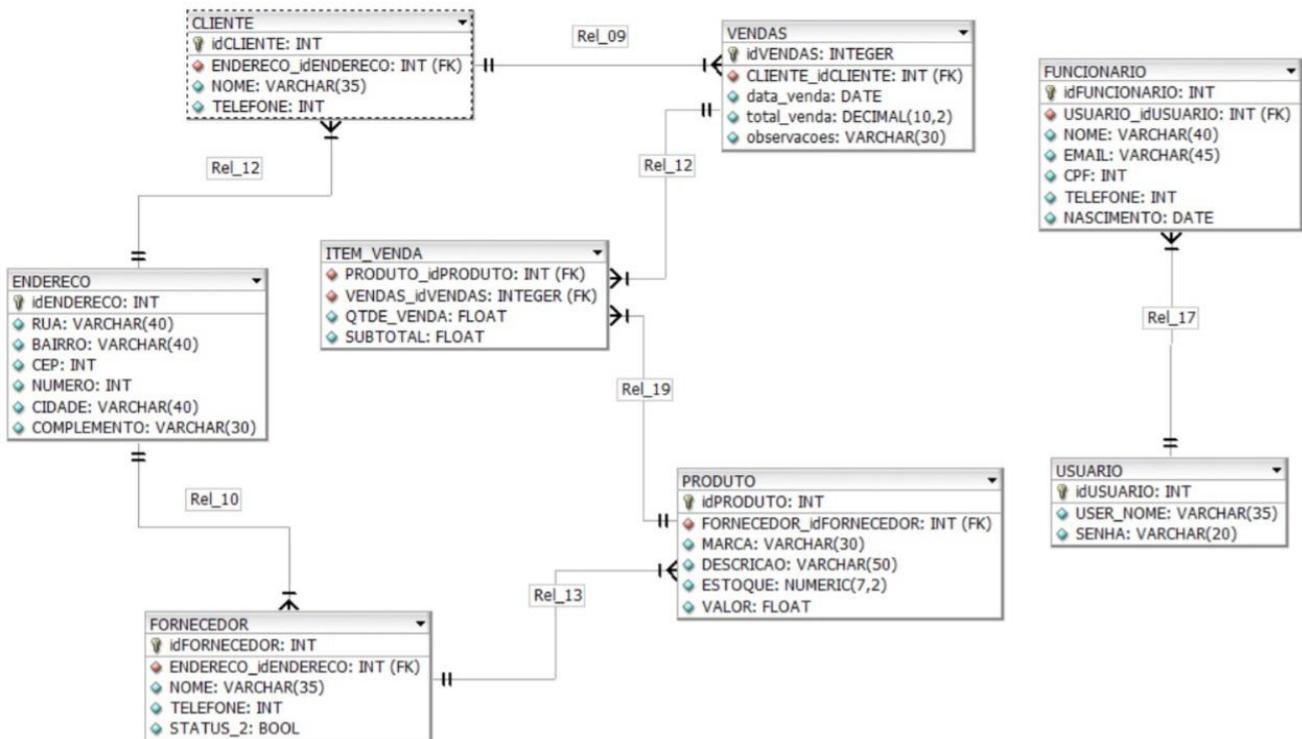


Figura 1 – Diagrama de Entidade e Relacionamento

4.2. DIAGRAMA DE CLASSES

Segundo IBM (2021), diagramas de classe são as cópias do sistema ou subsistema. Você pode utilizar os diagramas de classe para modelar os objetos que compõem o sistema, para exibir os relacionamentos entre os objetos e para descrever o que esses objetos fazem e os serviços que eles fornecem.

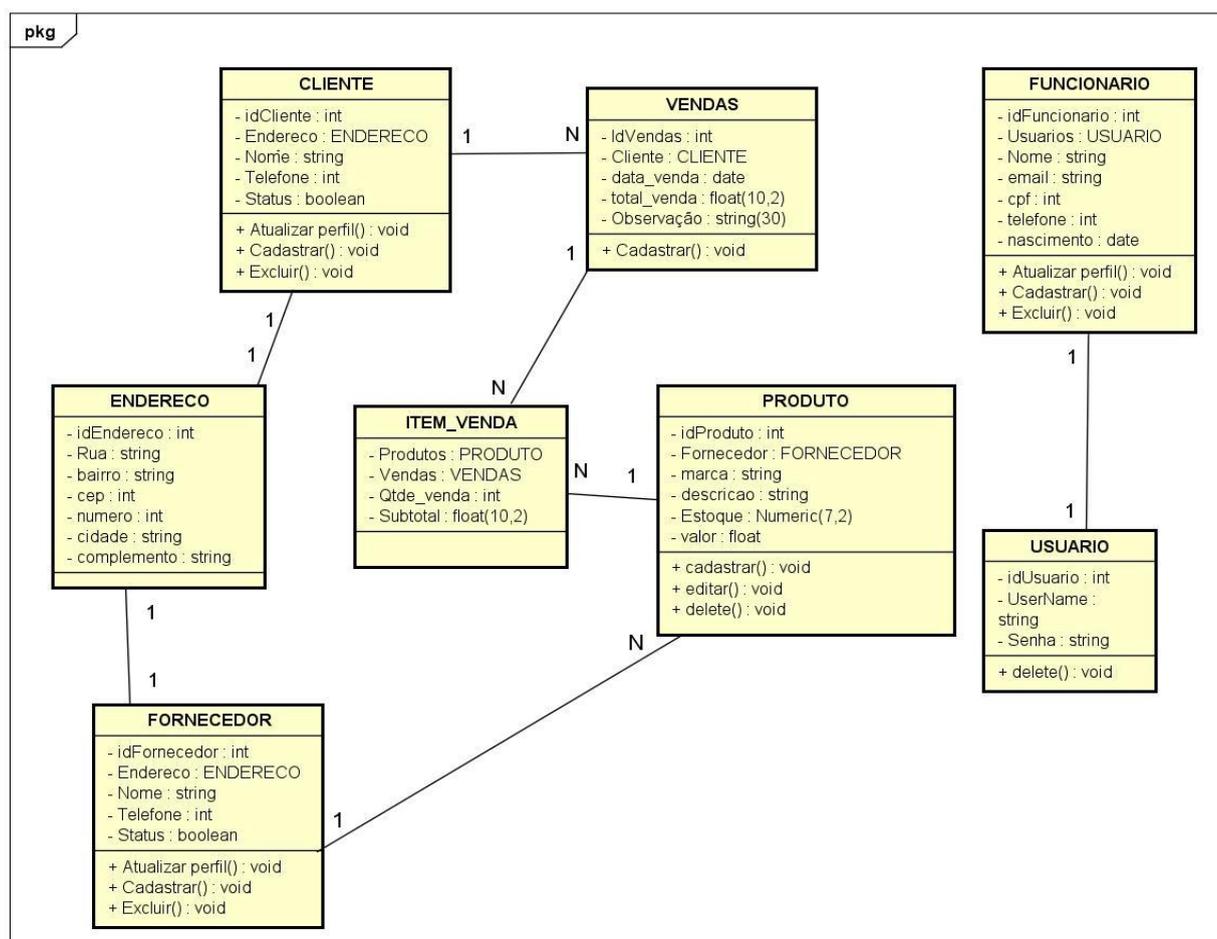


Figura 2 - Diagrama de Classes

4.3. DIAGRAMA DE CASOS DE USO

De acordo com IBM (2021), diagramas de casos de uso ilustram e definem o contexto e os requisitos de um sistema inteiro ou das partes importantes dele. É possível modelar um sistema complexo com um único diagrama de caso de uso ou ainda criar muitos diagramas de caso de uso para modelar os componentes do sistema. Normalmente, os diagramas de casos de uso são desenvolvidos nas fases iniciais de um projeto e são consultados em todo o processo de desenvolvimento.

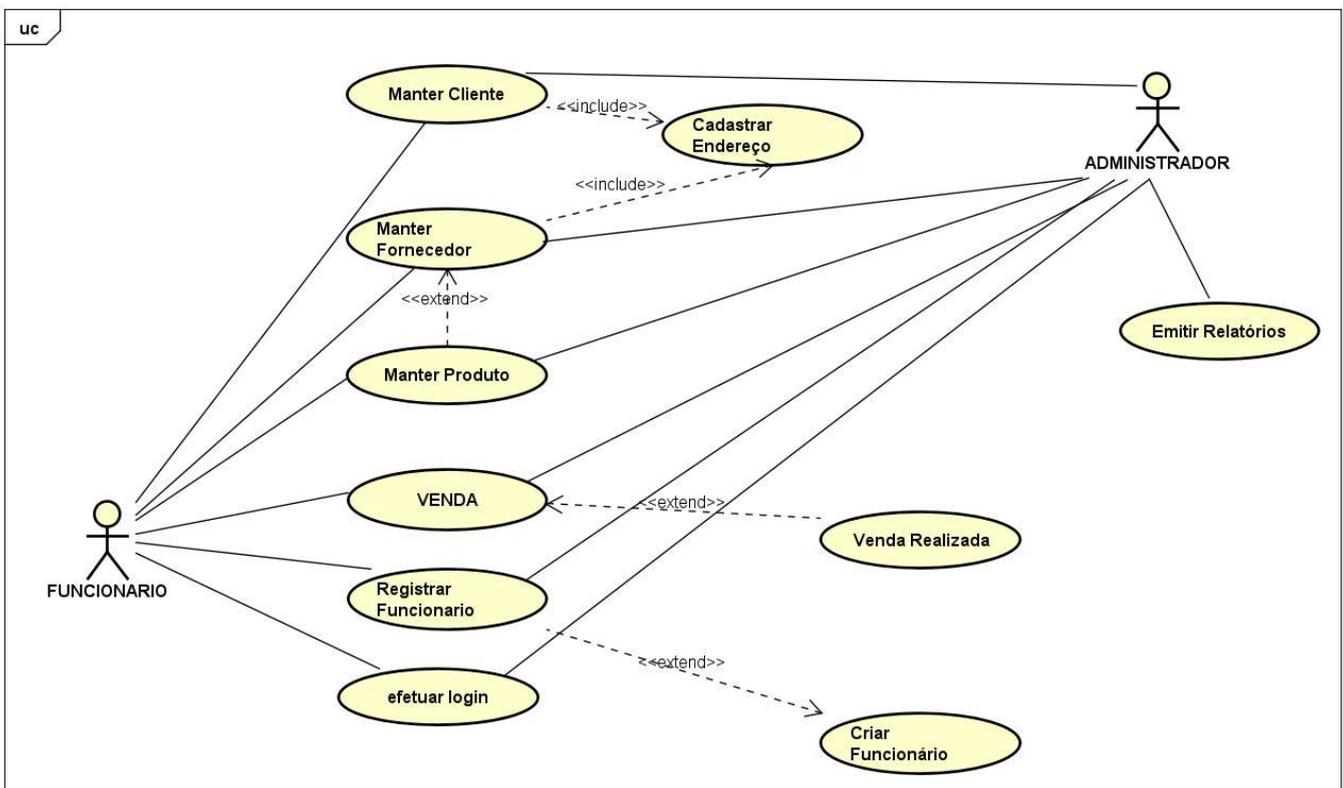


Figura 3 – Diagrama de Casos de Uso

5. DESENVOLVIMENTO DO TRABALHO

O desenvolvimento do trabalho foi realizado por etapas específicas, para que o sistema de gerenciamento fosse criado. Nas imagens a seguir, será detalhado o processo de criação do back-end.

5.1. ESTRUTURA DO BACKEND

5.1.1. CRIAÇÃO DO CLIENTE

O projeto foi construído pela *IDE* NetBeans, facilitando o processo de uma criação desktop por meio das figuras já prontas. A seguir o código para que o cliente seja cadastrado no sistema, com ligação ao banco de dados MySQL.

```
public class ClientesDAO {
    private Connection con;

    public ClientesDAO(){
        this.con = new ConnectionFactory().getConnection();
    }

    public void cadastrarCliente(Clientes obj){
        try {
            // 1° criar o comando SQL
            String sql = "insert into tb_clientes (nome,rg,cpf,email,telefone,celular,cep,endereco,numero,complemento,bairro,
                + "values (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?) ";

            // 2° conectar o banco de dados e organizar o comando sql
            PreparedStatement stmt = con.prepareStatement(sql);
            stmt.setString(1, obj.getNome());
            stmt.setString(2, obj.getRg());
            stmt.setString(3, obj.getCpf());
            stmt.setString(4, obj.getEmail());
            stmt.setString(5, obj.getTelefone());
            stmt.setString(6, obj.getCelular());
            stmt.setString(7, obj.getCep());
            stmt.setString(8, obj.getEndereco());
            stmt.setInt(9, obj.getNumero());
            stmt.setString(10, obj.getComplemento());

            stmt.setString(11, obj.getBairro());
            stmt.setString(12, obj.getCidade());
            stmt.setString(13, obj.getUF());

            // 3° Executar o comando sql
            stmt.execute();
            stmt.close();

            JOptionPane.showMessageDialog(null, "Cadastrado com Sucesso!");
        } catch (SQLException erro) {
            JOptionPane.showMessageDialog(null, "Erro:" + erro);
        }
    }
}
```

Figura 4 – Criação de cliente

A função de cadastro inicia-se a partir da string SQL, que recebe a instrução para cadastrar os dados no banco de dados, na tabela *tb_clientes*, com os parâmetros na ordem estabelecida. Em seguida, o *stmt* executa a instrução SQL. Por fim, valida-se se o cliente foi cadastrado com sucesso, e, em caso de erro, a falha é exibida na tela.

A conexão com o banco de dados é realizada a partir do *private connection con* aonde na linha a baixo a classe *ClientesDAO* recebe a conexão pela classe *ConnectionFactory*, e a partir da conexão o método para cadastrar o cliente é iniciado.

5.1.2. TELA PARA CADASTRO DE CLIENTE

Tela do software para cadastro de cliente

Cadastro de Clientes

Dados Pessoais Consulta de Clientes

Código:

Nome:

E-mail: Celular: Telefone:

CEP: Endereço: Nº:

Bairro: Cidade: Complemento: UF:

RG: CPF:

Cadastro de Clientes

Dados Pessoais Consulta de Clientes

Nome:

Código	Nome	RG	CPF	E-mail	Telefone	Celular	CEP	Endereço	Nº	Complemento	Bairro	Cidade	UF
4	Henrique da Silva	11.111.111-11	222.222.222-22	felipe@gmail.com	(11) 1 1111 - 1111	(18) 9 9739 - ...	19804-355	São José 446	466		Vila Souza	Assis	SP
5	José aparecido	11.111.111-11	222.222.222-22	felipe@gmail.com	() -	(18) 9 9739 - ...	19804-355	São José 446	466		Vila Souza	Assis	SP
6	João Vitor Varela	11.111.111-11	222.222.222-22	joaovitor@gmail...	() -	(18) 9 9739 - ...	19804-355	São José 446	466		Vila Souza	Assis	SP
7	Guilherme da Silva	11.111.111-11	222.222.222-22	guilhermesilva2...	(18) 3 3214 - 5200	(18) 9 9998 - ...	19804-510	Rua Santa Efr...	820	Loja	Vila Silvestre	Assis	SP
8	Antonio Silva	11.111.111-11	222.222.222-22	antoniosilva@g...	(18) 3 3215 - 5555	(18) 9 9820 - ...	19804-520	Rua Líbero Ba...	520	LOJA	Vila Maria Izabel	Assis	SP
9	Alex Sandro	12.345.678-90	123.456.789-20	alexsandro202...	(18) 3 3215 - 0520	(18) 9 9852 - ...	19804-530	Rua Santa Iz...	250	RESIDÊNCIA	Vila Maria Izabel	Assis	SP

Figura 5 – Tela de cadastro de cliente

A imagem relata a tela de cadastro de cliente, e a tela dos clientes cadastrados.

5.1.3. CRIAÇÃO DO FUNCIONÁRIO

O código para cadastro de funcionários no sistema.

```

public class FuncionariosDAO {

    //conexao BD
    private Connection con;

    public FuncionariosDAO() {
        this.con = new ConnectionFactory().getConnection();
    }

    // metodo para cadastrar funcionario
    public void cadastrarFuncionarios(Funcionarios obj) {

        try {

            // 1º criar o comando SQL
            String sql = "insert into tb_funcionarios (nome,rg,cpf,email,senha,cargo,nivel_acesso,telefone,celular,cep,ende
                + "values(?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)";

            // 2º conectar o banco de dados e organizar o comando sql
            PreparedStatement stmt = con.prepareStatement(sql);
            stmt.setString(1, obj.getNome());
            stmt.setString(2, obj.getRg());
            stmt.setString(3, obj.getCpf());
            stmt.setString(4, obj.getEmail());
            stmt.setString(5, obj.getSenha());
            stmt.setString(6, obj.getCargo());
            stmt.setString(7, obj.getNivel_acesso());
            stmt.setString(8, obj.getTelefone());

            stmt.setString(9, obj.getCelular());
            stmt.setString(10, obj.getCep());
            stmt.setString(11, obj.getEndereco());
            stmt.setInt(12, obj.getNumero());
            stmt.setString(13, obj.getComplemento());
            stmt.setString(14, obj.getBairro());
            stmt.setString(15, obj.getCidade());
            stmt.setString(16, obj.getUf());

            // 3º Executar o comando sql
            stmt.execute();
            stmt.close();

            JOptionPane.showMessageDialog(null, "Cadastrado com Sucesso!");

        } catch (SQLException erro) {
            JOptionPane.showMessageDialog(null, "Erro:" + erro);
        }
    }
}

```

Figura 6 - Criação de funcionário

A função para cadastro se inicia a partir da string SQL, a qual recebe a ação para cadastrar os dados no banco de dados, na tabela *tb_funcionarios*, com os devidos parâmetros. Após isso, o *stmt* executa a instrução SQL. E no final, verifica se o cliente foi cadastrado com sucesso, e, caso apresente algum erro, a falha é exibida na tela.

O método para cadastrar o funcionário, se estendeu da classe CLIENTES, mas o funcionário recebeu 3 atributos a mais, que seria o cargo, o nível de acesso e a senha.

5.1.4. TELA PARA CADASTRO DE FUNCIONÁRIO

Tela do sistema para que o funcionário seja cadastrado.

Cadastro de Funcionários

Dados Pessoais Consulta de funcionários

Código:

Nome:

E-mail: Celular: Telefone:

CEP: Endereço: Nº:

Bairro: Cidade: Complemento: UF:

RG: CPF: Cargo:

Senha: Nível de acesso:

Cadastro de Funcionários

Dados Pessoais Consulta de funcionários

Nome:

Código	Nome	RG	CPF	E-mail	Senha	Cargo	Nível Acesso	Telefone	Celular	CEP	Endereço	Nº	Comple...	Bairro	Cidade	UF
5	João Vitor	00.000.0...	123.456.789...	joaovito...	123	Funcionário	Usuário	(18) 9 9...	(18) 9 9...	19804-...	Rua São José	446	casa	Vila So...	Assis	SP

Figura 7 – Tela de cadastro de funcionário

A imagem apresenta a tela para o cadastro, e também mostra a tela para a consulta dos funcionários após o cadastramento, com todas as informações de cadastro.

O sistema tem outras telas de cadastro, relatórios e menu.

6. CONCLUSÃO

A construção do sistema de gerenciamento de pesca, se torna uma ferramenta tecnológica eficiente, contribui para uma melhor logística da empresa, e auxilia a gestão do comércio. O software fornecido facilita a gestão das vendas, a entrada e saída de produtos e o histórico das vendas.

Com o sistema, a empresa obtém uma boa experiência para ter controle sobre suas operações, desde o cadastro dos produtos, funcionários, clientes, até o gerenciamento das vendas. Isso conclui em uma maior eficiência nas atividades, redução significativa de erros em comparação ao processo manual, uma melhor tomada de decisão, além de otimizar o dia a dia da empresa.

6.1. TRABALHOS FUTUROS

O software ajuda o atendimento, e a gestão da empresa como um todo, mas também tem uma grande capacidade para aprimorações e melhorias, como a análise de dados em tempo real, disponibilizar dados mais detalhados de comportamento de vendas e o estoque. A inserção do sistema para dispositivos móveis, pode ser uma área interessante para ser desenvolvida, buscando uma maior acessibilidade e controle das operações. Além de todas essas evoluções, expandir o sistema e conectá-lo à uma plataforma de pagamentos online, poderia conceder uma melhor qualidade e valor a aplicação, com a possibilidade dos clientes avaliarem o aplicativo com feedbacks, transformando a aplicação mais completa e eficiente para a empresa.

REFERÊNCIAS

Bessa, André Bessa, Java: o que é, linguagem e um Guia para iniciar na tecnologia. Alura, 2023. Disponível em: <https://www.alura.com.br/artigos/java?srsIid=AfmBOorZa8LOyjXAv5HEM0KrGXjym_6VTizjyn4Xmhu4RHn8dPUsNf1E>. Acesso em: 07 ago. 2024.

Cardoso, Rodrigo Cardoso, MySQL: guia completo de como utilizar o sistema. LocaWeb, 2023. Disponível em: <<https://www.locaweb.com.br/blog/temas/codigo-aberto/mysql-guia-completo-de-como-utilizar-o-sistema/>>. Acesso em: 07 ago. 2024.

Diagramas de Caso de Uso. IBM, 2021. Disponível em: <<https://www.ibm.com/docs/pt-br/rsm/7.5.0?topic=diagrams-use-case>>. Acesso em: 27 mar. 2024.

Diagrama de Classes. IBM, 2021. Disponível em: <<https://www.ibm.com/docs/pt-br/rsas/7.5.0?topic=structure-class-diagrams>>. Acesso em: 28 mar. 2024.

Gestão Click. Programa para loja de pescaria. São Paulo. Disponível em: <<https://gestaoclick.com.br/programa-para-loja-de-pescaria/>>. Acesso em: 19 jun. 2023.

Java Swing: a biblioteca para criação de interfaces gráficas. Awari, 2023. Disponível em: <<https://awari.com.br/java-swing/>>. Acesso em: 07 ago. 2024.

Modelos softwares com Astah Community. TechTudo, 2022. Disponível em: <<https://www.techtudo.com.br/tudo-sobre/astah-community/>>. Acesso em: 07 ago. 2024.

Modelos e Diagramas UML. IBM, 2021. Disponível em: <<https://www.ibm.com/docs/pt-br/rsas/7.5.0?topic=models-uml-diagrams>>. Acesso em: 07 ago. 2024.

Oliveira, Hewerton Luís Pereira Santiago, João Antonio Oliveira Pedrosa, Luiz Henrique Rodrigues Silva, Marcelo Caramuru Pimentel Fraga, Miguel Rodrigues Guimarães de Oliveira. TechFish - Software de Auxílio a Gerência de Ambientes Pesque e Pague. 2018. 16p. Monografia - CEFET-MG - Centro Federal de Educação Tecnológica de Minas Gerais, Minas Gerais, 2018.

Oliveira, Sheilane Karla Martins de Oliveira. Proposta de Requisitos de Software para uma loja de móveis: uma articulação entre computação, gestão e ciência da informação. 2017. 48p. Trabalho de Conclusão de Curso – Centro de Artes e Comunicação. Universidade federal de Pernambuco, Recife, 2017.

O papel do Diagrama Entidade-Relacionamento na modelagem de dados. Awari, 2023. Disponível em: <<https://awari.com.br/diagrama-entidade-relacionamento/>>. Acesso em: 28 mar. 2024.