

**UNIVERSIDADE ESTADUAL DA PARAÍBA**

**DANILO PEDRO DA SILVA VALÉRIO**

**Documento de Visão do VL Store**

**Campina Grande - PB**

**2025**

**Danilo Pedro da Silva Valério**

## **Documento de Visão do VL Store**

**Este trabalho, intitulado "Documento de Visão do VL Store", é apresentado ao curso de Ciência da Computação, como parte dos requisitos necessários à obtenção da nota da disciplina de Programação Web, visando o desenvolvimento de um site funcional e de qualidade.**

**Professor:** Allan Villar de Carvalho

**Campina Grande - PB**

**2025**

## 1. Resumo:

O avanço das tecnologias web tem impulsionado a criação de marketplaces modernos, capazes de oferecer soluções eficientes para gestão de produtos e vendedores. Este trabalho apresenta o desenvolvimento de um sistema de marketplace simplificado, focado na usabilidade e na escalabilidade. A aplicação será construída utilizando Node.js com Express.js no back-end, React com Next.js e Bootstrap no front-end, além do banco de dados MongoDB para armazenar as informações. O sistema permite o gerenciamento de lojas, produtos e compradores, garantindo uma experiência dinâmica e intuitiva. A adoção da arquitetura MVC possibilitou uma organização modular do código, favorecendo a manutenção e evolução da aplicação. Além disso, foram implementadas boas práticas de desenvolvimento web, como a criação de APIs RESTful para comunicação entre as camadas do sistema. Com isso, o projeto busca demonstrar a

**Palavras-chave:** Marketplace, Desenvolvimento Web, Node.js, Express.js, React, Next.js, Bootstrap, MongoDB.

### **Abstract**

This work presents the development of a simplified marketplace using an architecture based on Node.js with Express.js for the back-end, React with Next.js and Bootstrap for the front-end, and MongoDB as the database. The system aims to provide an efficient platform for managing purchases and sales, allowing the registration, management, and visualization of products, stores, and buyers. The application follows the MVC architectural pattern, ensuring the separation between the user interface, business logic, and persistence layer. Additionally, best development practices are adopted, such as code modularization and the use of RESTful APIs for communication between the front-end and back-end. The choice of technologies seeks to provide scalability, flexibility, and a responsive user experience.

**Keywords:** Marketplace, Store Management, Commercial Management, Node.js, Express.js, React, Next.js, Bootstrap, MongoDB, MVC, Web Development.

## 2. Descrição do Sistema

O vl-store é um software de gestão comercial projetado para auxiliar no gerenciamento de produtos, funcionários, vendas e caixas. O sistema permite um controle abrangente do estoque, transações financeiras e desempenho dos vendedores, garantindo que o gestor tenha total visibilidade e controle sobre o seu negócio.

## 3. Requisitos funcionais

### 4. Nome: Gerenciamento de Produtos

#### Descrição:

**Esta funcionalidade possibilita o gerenciamento completo do cadastro de produtos, permitindo operações de CRUD (Criar, Ler, Atualizar e Deletar) com acesso restrito conforme o perfil do usuário (apenas administradores têm permissão para alterações).**

#### a. Cadastrar Produtos

##### Descrição:

Permite a inclusão de novos produtos no sistema com os seguintes atributos:

- **Título:** Nome ou identificação do produto.
- **Categoria:** Classificação do produto para organização e busca.
- **Material:** Informação sobre a composição ou material do produto.
- **Referência:** Código numérico de 4 dígitos gerado aleatoriamente e que não poderá ser alterado após o cadastro.
- **Gênero:** Campo opcional para segmentação (ex.: masculino, feminino ou unissex).
- **Variações:** Cada produto pode ter uma ou mais variações, onde cada variação deve conter:

- **Descrição da variação:** Detalhes específicos (ex.: cor, tamanho, modelo).
- **Quantidade:** Estoque disponível para a variação.
- **Valor:** Preço aplicado à variação.

*Permissões:*

- **Administrador (usuário gerente):** Exclusivo para o cadastro de produtos.

## **b. Leitura dos Produtos**

**Descrição:**

Exibição dos produtos cadastrados em uma tabela interativa que possibilita:

- **Busca:** Pesquisa por título, categoria ou referência.
- **Visualização Detalhada:** Exibição completa dos dados do produto, incluindo suas variações, ao selecionar um item.

*Permissões:*

- **Administrador:** Acesso completo.
- **Vendedor:** Pode realizar consulta, mas sem acesso às funcionalidades de edição ou exclusão.

## **c. Atualizar Produtos**

**Descrição:**

Permite a edição dos dados de um produto existente, com a ressalva de que o campo **Referência** é imutável.

- **Campos Editáveis:** Título, Categoria, Material, Gênero e informações das variações (descrição, quantidade e valor).
- **Processo:** A atualização deve ser feita a partir da janela de visualização detalhada do produto, garantindo que as alterações sejam salvas de forma consistente.

*Permissões:*

- **Administrador:** Exclusivo para realizar atualizações.

#### **d. Deletar Produtos**

##### **Descrição:**

Habilita a exclusão de um produto do sistema, removendo todas as informações associadas, inclusive suas variações.

- **Confirmação de Exclusão:** Antes da remoção, o sistema deve solicitar uma confirmação para evitar exclusões acidentais.

##### *Permissões:*

- **Administrador:** Exclusivo para a exclusão de produtos.

#### **5. Nome: Controle de Funcionários**

##### **Descrição:**

Esta funcionalidade gerencia o cadastro e as informações dos funcionários, permitindo operações de CRUD (Criar, Ler, Atualizar e Deletar) e a visualização de dados complementares (como vendas e vales), com acesso restrito ao administrador.

##### **a. Cadastrar Funcionário**

##### **Descrição:**

Permite o registro de novos funcionários com os seguintes dados:

- **Nome:** Nome completo do funcionário.
- **Cargo:** Posição ou função desempenhada na empresa.
- **Data de Nascimento:** Informação necessária para controle e conformidade com políticas internas.
- **Telefone:** Contato para comunicação.
- **CPF:** Identificação única do funcionário.

##### *Permissões:*

- **Administrador (usuário gerente):** Exclusivo para o cadastro de funcionários.

## **b. Leitura dos Funcionários**

### **Descrição:**

Exibição dos funcionários cadastrados em uma tabela que permite:

- **Busca e Filtro:** Pesquisa por nome, cargo ou CPF.
- **Visualização Detalhada:** Ao selecionar um funcionário, o sistema apresenta todas as informações cadastradas e, adicionalmente, dados sobre vendas e vales – dados esses que são acessíveis somente ao administrador.

### *Permissões:*

- **Administrador:** Visualiza todas as informações detalhadas e históricas.
- **Outros Perfis:** Geralmente têm acesso apenas a informações não sensíveis ou para fins de consulta.

## **c. Atualizar Funcionários**

### **Descrição:**

Facilita a edição das informações de um funcionário já cadastrado.

- **Campos Editáveis:** Nome, Cargo, Data de Nascimento, Telefone e outros dados que possam precisar de atualização.
- **Restrições:** Dados únicos (como CPF) podem ser definidos como imutáveis, conforme política do sistema.

### *Permissões:*

- **Administrador:** Exclusivo para a realização de atualizações.

## **d. Deletar Funcionários**

### **Descrição:**

Permite a remoção do cadastro de um funcionário do sistema.

- **Confirmação:** Deve haver um mecanismo de confirmação para evitar a exclusão acidental dos dados.

*Permissões:*

- **Administrador:** Exclusivo para a exclusão de funcionários.

## 6. Nome: Gerenciamento de Vendas

### **Descrição:**

Funcionalidade que permite registrar e gerenciar as vendas realizadas, com permissões diferenciadas para vendedores (que podem registrar e visualizar apenas suas vendas) e administradores (com acesso completo a todas as operações).

#### **a. Registrar Venda**

##### **Descrição:**

Permite o registro de uma nova venda. Ao registrar, o sistema deverá capturar os seguintes dados:

- **Código da venda:** Identificador único para a transação.
- **Lista de produtos vendidos:** Incluindo identificação, quantidade e valor unitário de cada produto.
- **Valor total:** Soma dos valores dos produtos, aplicando possíveis descontos ou acréscimos.
- **Forma de pagamento:** Opções como dinheiro, cartão ou outros métodos.
- **Vendedor responsável:** Identificação do usuário que registrou a venda.
- **Data e hora da transação.**

*Permissões:*

- **Vendedor:** Pode registrar e visualizar suas próprias vendas.
- **Administrador:** Pode registrar, visualizar, atualizar e deletar qualquer venda, se necessário.



## **b. Visualizar Vendas**

### **Descrição:**

Exibe uma lista de todas as vendas registradas em formato de tabela, com possibilidade de:

- **Filtrar por data, vendedor e forma de pagamento.**
- **Ordenar as vendas por data ou valor.**

### *Permissões:*

- **Vendedor:** Visualiza apenas as vendas registradas por ele.
- **Administrador:** Visualiza todas as vendas.

## **c. Detalhar Venda**

### **Descrição:**

Ao selecionar uma venda na listagem, o sistema apresentará uma visão detalhada com:

- **Informações completas dos produtos vendidos (descrição, quantidade, valor unitário).**
- **Resumo da transação (valor total, forma de pagamento e dados do vendedor).**

### *Permissões:*

- **Vendedor:** Detalhamento de suas próprias vendas.
- **Administrador:** Detalhamento de todas as vendas.

## **d. Atualizar Venda**

### **Descrição:**

Permite a edição de informações de uma venda já registrada, em casos de correções ou ajustes, respeitando as regras de integridade do sistema.

- **Campos editáveis:** Itens que possam ter sido registrados incorretamente, como forma de pagamento ou quantidades (sujeito a políticas internas).
- **Histórico de alterações:** Manutenção de logs para auditoria.

*Permissões:*

- **Administrador:** Pode realizar atualizações em qualquer venda.
- **Vendedor:** Geralmente não tem permissão para editar, exceto em situações previamente definidas.

**e. Deletar Venda**

**Descrição:**

Possibilita a exclusão de uma venda, com a devida confirmação para evitar exclusões acidentais.

- **Confirmação:** Janela de confirmação antes da exclusão.
- **Histórico:** Registro da exclusão para auditoria, se aplicável.

*Permissões:*

- **Administrador:** Exclusão autorizada.
- **Vendedor:** Normalmente não autorizado a deletar vendas.

**7. Nome: Controle de Caixa**

**Descrição:**

Funcionalidade que gerencia as movimentações financeiras diárias, registrando entradas e saídas, rastreando transações e permitindo a geração de relatórios financeiros.

**a. Registrar Entrada de Caixa**

**Descrição:**

Permite o registro de todas as entradas financeiras, provenientes de:

- **Vendas realizadas.**
- **Depósitos ou outros recebimentos.**

*Campos obrigatórios:*

- **Valor recebido.**
- **Data e hora.**
- **Descrição da origem da entrada.**
- **Identificação do responsável pela entrada.**

*Permissões:*

- **Administrador:** Efetua e revisa os registros de entradas.

#### **b. Registrar Saída de Caixa**

**Descrição:**

Permite o registro de todas as saídas financeiras, como:

- **Despesas operacionais.**
- **Pagamentos diversos.**

*Campos obrigatórios:*

- **Valor da saída.**
- **Data e hora.**
- **Descrição detalhada da despesa.**
- **Identificação do responsável.**

*Permissões:*

- **Administrador:** Realiza o registro e supervisão das saídas.

#### **c. Visualizar Movimentações do Caixa**

**Descrição:**

Exibe uma tabela consolidada de todas as movimentações (entradas e saídas) com possibilidade de:

- **Filtrar por data, tipo (entrada/saída) e responsável.**
- **Ordenar as transações para análise rápida.**

*Permissões:*

- **Administrador:** Acesso completo à visualização e análise.

#### **d. Fechamento Diário do Caixa**

**Descrição:**

Permite o fechamento do caixa ao final do dia, consolidando os totais de:

- **Entradas.**
- **Saídas.**
- **Saldo final.**

*Funcionalidades adicionais:*

- **Geração de um relatório resumido diário para conferência e auditoria.**

*Permissões:*

- **Administrador:** Realiza o fechamento diário e tem acesso ao relatório.

#### **e. Ajuste de Movimentações**

**Descrição:**

Em casos de erros ou necessidade de correção, o sistema possibilitará ajustes nas movimentações registradas, garantindo:

- **Registro do histórico de alterações para auditoria.**
- **Validação dos ajustes realizados.**

*Permissões:*

- **Administrador:** Exclusivo para realizar ajustes no caixa.

### **8. Nome: Relatórios**

**Descrição:**

Funcionalidade que gera e disponibiliza diversos relatórios para análise de desempenho comercial, financeiro e operacional, permitindo uma visão estratégica do negócio.

#### **a. Relatório de Vendas**

**Descrição:**

Gera relatórios detalhados das vendas realizadas, com possibilidades de:

- **Filtragem por período (diário, semanal, mensal).**
- **Segmentação por forma de pagamento, produtos e vendedores.**
- **Exibição de totais, número de transações e produtos mais vendidos.**

*Permissões:*

- **Administrador:** Acesso completo.
- **Vendedor:** Pode acessar somente os dados referentes às suas vendas, se aplicável.

#### **b. Relatório de Desempenho de Vendedores**

**Descrição:**

Apresenta indicadores de performance dos vendedores, incluindo:

- **Total de vendas por vendedor.**
- **Média de vendas diária/por período.**
- **Ranking dos melhores vendedores.**

*Permissões:*

- **Administrador:** Visualização completa para análise e tomada de decisão.

#### **c. Relatório Financeiro**

**Descrição:**

Consolida os dados do Controle de Caixa, apresentando:

- **Resumo das entradas e saídas.**
- **Saldo final do período.**
- **Gráficos e comparativos para análise do fluxo de caixa.**

*Permissões:*

- **Administrador:** Acesso para análise financeira detalhada.

#### **d. Relatório Personalizado**

**Descrição:**

Permite ao administrador definir filtros e parâmetros para a geração de relatórios sob medida, como:

- **Análise de desempenho por período específico.**
- **Comparativo entre diferentes períodos ou setores.**

- **Segmentação por campanhas promocionais ou categorias de produtos.**

*Permissões:*

- **Administrador:** Exclusivo para criação e visualização de relatórios customizados.

#### **e. Exportação e Compartilhamento de Relatórios**

**Descrição:**

Oferece a possibilidade de exportar os relatórios gerados para formatos como:

- **PDF.**

*Funcionalidades adicionais:*

- **Facilitar o compartilhamento e a análise externa dos dados.**

*Permissões:*

- **Administrador:** Responsável por exportar e compartilhar os relatórios conforme necessidade.

#### **9. Nome: Acesso ao software como visitante.**

**Descrição:**

Acesso ao funcionamento do software como visitante, sendo assim, qualquer pessoa pode acessar as funcionalidades do sistema e fazer alterações, todavia, estas alterações não serão efetivadas. Deverá ser feita para que o usuário visitante consiga ver o funcionamento do software antes de se cadastrar. Além disso, dados fictícios serão fixados nesta modalidade de acesso.

*Permissões:*

**Visitante:** Poderá acessar o sistema e visualizar dados fictícios para ter uma noção do funcionamento do software sem precisar efetuar cadastro.

## **10. Requisitos não funcionais**

### **Acessibilidade**

#### **Descrição:**

O sistema deve ser estilizado com um padrão de alto contraste, garantindo melhor visibilidade para usuários com deficiência visual. Esse padrão também deve ser aplicado na exportação de relatórios.

#### **CrITÉrios de Aceitação:**

- Todos os elementos da interface seguem o padrão de alto contraste.
- Os relatórios exportados mantêm o padrão de alto contraste para melhor legibilidade.

### **Formatação de Senhas**

#### **Descrição:**

O sistema deve impor regras para a criação de senhas, garantindo que tenham entre 8 e 16 caracteres e incluam pelo menos uma letra maiúscula, uma letra minúscula, um número e um caractere especial.

#### **CrITÉrios de Aceitação:**

- O sistema exibe um erro ao tentar cadastrar uma senha fora das regras estabelecidas.
- O usuário só pode prosseguir com o cadastro se a senha atender a todas as regras.
- A regra de formatação da senha é informada ao usuário antes da criação da conta.

### **Segurança (Criptografia de Senhas)**

#### **Descrição:**

As senhas dos usuários devem ser criptografadas utilizando o algoritmo Bcrypt antes de serem armazenadas no banco de dados.

#### **CrITÉrios de Aceitação:**

- As senhas nunca são armazenadas em formato de texto puro.
- O sistema utiliza Bcrypt com um fator de custo adequado para garantir segurança.
- O processo de login compara a senha inserida com o hash armazenado.

## **Desempenho**

### **Descrição:**

O sistema deve armazenar em cache as consultas mais frequentes para otimizar a velocidade de resposta.

### **CrITÉrios de Aceitação:**

- As consultas frequentes devem ser armazenadas em cache para reduzir a carga no banco de dados.
- O tempo de resposta para essas consultas deve ser menor quando acessadas novamente.
- O cache deve ser atualizado periodicamente para garantir dados consistentes.

## **11. Perfil dos Usuários**

### **11.1. Administrador(Gerente contratador do serviço)**

- **Descrição:** Usuário responsável pelo gerenciamento completo do sistema, tendo acesso total às funcionalidades.
- **Tipo:** Gerente ou proprietário do comércio.
- **Responsabilidade:**
  - Cadastrar, visualizar, editar e excluir produtos.
  - Gerenciar funcionários, incluindo cadastro, atualização e exclusão.
  - Acompanhar vendas e monitorar o desempenho da equipe.
  - Controlar movimentações de caixa e gerar relatórios financeiros.



### 11.2. Vendedor

- **Descrição:** Funcionário responsável por realizar vendas e auxiliar no controle de estoque.
- **Tipo:** Funcionário com permissões limitadas.
- **Responsabilidade:**
  - Registrar vendas no sistema.
  - Consultar produtos disponíveis para informar clientes.
  - Acompanhar seu próprio desempenho de vendas.

### 11.3 Visitante

- **Descrição:** Usuário não autenticado que pode navegar no sistema sem permissões de alteração.
- **Tipo:** Cliente ou interessado em ver o funcionamento do software.
- **Responsabilidade:**
  - Pesquisar e visualizar produtos disponíveis.
  - Obter informações sobre categorias e preços.

## 12. Arquitetura:

O projeto será desenvolvido seguindo a arquitetura MVC (Model-View-Controller), garantindo uma separação clara entre as responsabilidades de apresentação, lógica de negócio e manipulação de dados. Essa abordagem facilita a manutenção, a escalabilidade e a organização do código.

Além disso, o projeto será construído utilizando um paradigma orientado a objetos (POO), garantindo reutilização de código, modularidade e encapsulamento. As classes serão projetadas para representar entidades e suas interações de forma estruturada.

A implementação será feita em uma linguagem tipada, garantindo maior segurança e previsibilidade no desenvolvimento,

reduzindo erros relacionados a tipos de dados. Isso também facilita a detecção de inconsistências durante a fase de compilação ou interpretação.

### **13. Classes:**

#### **a. Modelos (Model)**

##### **■ Arquivo: Produto.ts**

- **Classe: Produto**
- **Responsabilidade:** Representar um produto com atributos de nome, categoria, material, referência, variações, cada variação tendo sua descrição curta, quantidade e valor.

##### **■ Arquivo: Funcionario.ts**

- **Classe: Funcionario**
- **Responsabilidade:** Representar um funcionário com dados como nome, cargo, telefone, CPF, e outros. Relacionamento com vendas realizadas (caso o vendedor esteja associado à venda).

##### **■ Arquivo: Venda.ts**

- **Classe: Venda**
- **Responsabilidade:** Representar uma venda, incluindo produtos vendidos, valor total, forma de pagamento, vendedor responsável e a data da transação.

##### **■ Arquivo: Usuario.ts**

- **Classe: Usuario**
- **Responsabilidade:** Representar o usuário, incluindo nome completo, senha criptografada, e-mail de recuperação de senha, telefone e data de nascimento.

## **b. Controladores (Controller)**

- **Arquivo: ProdutoController.ts**
  - **Classe: ProdutoController**
  - **Responsabilidade: Gerenciar as operações de CRUD para os produtos (criar, listar, atualizar, excluir produtos).**
  
- **Arquivo: FuncionarioController.ts**
  - **Classe: FuncionarioController**
  - **Responsabilidade: Gerenciar as operações de CRUD para os funcionários (criar, listar, atualizar, excluir funcionários).**
  
- **Arquivo: VendaController.ts**
  - **Classe: VendaController**
  - **Responsabilidade: Gerenciar as operações de CRUD para as vendas (registrar, listar, visualizar detalhes, atualizar e excluir vendas).**
  
- **Arquivo: UsuarioController.ts**
  - **Classe: UsuarioController**
  - **Responsabilidade: Gerenciar as operações de CRUD para as vendas (registrar, listar, visualizar detalhes, atualizar e excluir vendas).**

## **c. Rotas (Routes)**

- **Arquivo: produtoRoutes.js**
  - **Classe: ProdutoRoutes**
  - **Responsabilidade: Define as rotas que estarão disponíveis para interação com o modelo Produto**

(e.g., /api/produtos para criar, listar, editar e excluir produtos).

- **Arquivo: funcionarioRoutes.js**
  - **Classe: FuncionarioRoutes**
  - **Responsabilidade:** Define as rotas para manipulação de dados de funcionários (e.g., /api/funcionarios para criar, listar, editar e excluir funcionários).
  
- **Arquivo: vendaRoutes.js**
  - **Classe: VendaRoutes**
  - **Responsabilidade:** Define as rotas para operações relacionadas às vendas (e.g., /api/vendas para registrar, listar e atualizar vendas).
  -
  
- **Arquivo: usuarioRoutes.js**
  - **Classe: usuarioRoutes**
  - **Responsabilidade:** Define as rotas para operações relacionadas aos usuarios (e.g., /api/user para registrar, listar e atualizar usuários).

#### **d. Visão (View) (Front-End)**

- **Pages:**
  - **Arquivo: Produtos.tsx**
    - **Responsabilidade:** Componente tsx. para a interface de usuário relacionada à exibição, criação, edição e exclusão de produtos.
  
  - **Arquivo: Funcionarios.tsx**
    - **Responsabilidade:** Componente tsx para a interface de usuário relacionada à exibição e gerenciamento de funcionários.
  
  - **Arquivo: Vendas.tsx**

- **Responsabilidade:** Componente `tsx.` para a interface de usuário relacionada ao gerenciamento de vendas, incluindo visualização e registro de transações.
- **Arquivo: Relatórios.tsx**
  - **Responsabilidade:** Componente `tsx.` para a interface de usuário relacionada ao gerenciamento de relatórios.
- **Arquivo: Caixa.tsx**
  - **Responsabilidade:** Componente `tsx.js` para a interface de usuário relacionada ao caixa, incluindo visualização e registro de transações.
- **Arquivo: Menu.tsx**
  - **Responsabilidade:** Componente `tsx.` para a interface de usuário relacionada ao menu principal do sistema no qual o usuário pode ir para as demais páginas.

#### 14. Tecnologias Utilizadas

O **vi-store** será desenvolvido utilizando as seguintes tecnologias:

- **Backend:** Node.js com Express.js para criação da API REST + Typescript.
- **Banco de Dados:** MongoDB para armazenamento das informações.
- **Frontend:** React.js + next.js para a interface do usuário além do bootstrap para auxiliar na responsividade em diferentes telas.
- **Autenticação:** JWT (JSON Web Token) para controle de acesso.
- **Arquitetura:** MVC (Model-View-Controller) seguindo paradigma orientado a objetos.
- **Segurança:** Bcrypt para hash de senhas e validações no backend.
- **Versionamento:** GitHub.