MVP - Catalogo de Produtos

Manual Técnico de Software

Histórico de Revisão

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** | **Revisor** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

INDICE

[1 Introdução 4](#_Toc170889341)

[1.1 Objetivo 4](#_Toc170889342)

[1.2 Visão Geral 4](#_Toc170889343)

[2 Ambiente de Desenvolvimento 5](#_Toc170889344)

[2.1 Ferramenta 5](#_Toc170889345)

[2.2 Estrutura da Solução 5](#_Toc170889346)

[2.3 SGBD 7](#_Toc170889347)

[3 Requisitos 7](#_Toc170889348)

[4 Banco de Dados 7](#_Toc170889349)

[5 APIs 8](#_Toc170889350)

[6 Funcionamento 9](#_Toc170889351)

[6.1 Frontend 9](#_Toc170889352)

[6.2 Backend 9](#_Toc170889353)

[7 Interface do Usuário 9](#_Toc170889354)

[7.1 Login com Documento 10](#_Toc170889355)

[7.2 Carrosel/Slider de Produtos 10](#_Toc170889356)

[7.3 Catalogo de Produtos 11](#_Toc170889357)

[7.4 Detalhes do Produto 11](#_Toc170889358)

[7.5 Checkout 12](#_Toc170889359)

[7.6 Comprovante 12](#_Toc170889360)

[8 Manutenção e Atualizações 12](#_Toc170889361)

[9 Pendencias 12](#_Toc170889362)

[10 Aprovação 12](#_Toc170889363)

# Introdução

Este manual técnico foi elaborado para fornecer uma compreensão detalhada do software. Ele destina-se a desenvolvedores, abordando desde a instalação até a solução de problemas. Com este guia, espera-se facilitar o uso eficiente e seguro das funcionalidades do software, garantindo um suporte contínuo e eficaz.

## Objetivo

O objetivo deste documento é oferecer orientações claras e detalhadas sobre o desenvolvimento e uso do software web de “Catalogo de Produtos”. Ele tem por finalidade descrever o processo básico para o desenvolvimento de software de acordo com o conceito de boas práticas e melhorias estabelecidos pela empresa Samsung.

## Visão Geral

O software web de Catalogo de Produtos oferece uma solução abrangente de um e-commerce instalado em um totem de atendimento. Ele é ideal para soluções que buscam otimizar as vendas de produtos. Através de um input básico do sistema e possível escolher os produtos a serem vendidos. Essa flexibilidade permite uma personalização completa do sistema, atendendo às necessidades específicas de cada negócio.

# Ambiente de Desenvolvimento

## Ferramenta

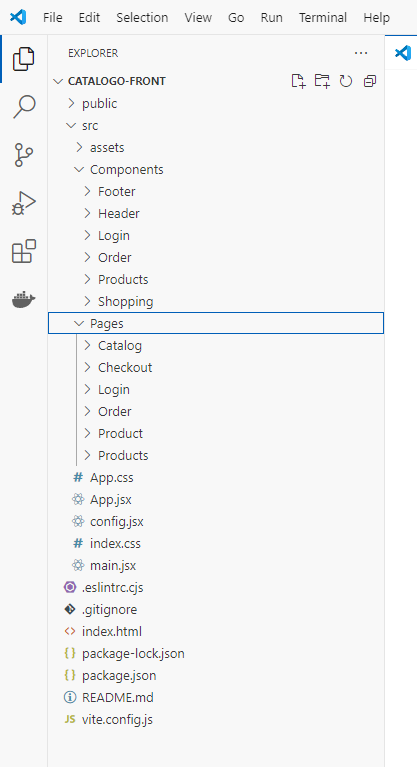
Para o desenvolvimento do software web, será utilizado o Intellij e Visual Studio Code como ferramentas principais, garantindo alta performance, segurança e compatibilidade multiplataforma. Eles permitem que o software funcione de maneira eficiente em diferentes sistemas operacionais, como Windows, MacOS e Linux, proporcionando uma experiência consistente e otimizada para todos os usuários.

|  |  |
| --- | --- |
|  |  |

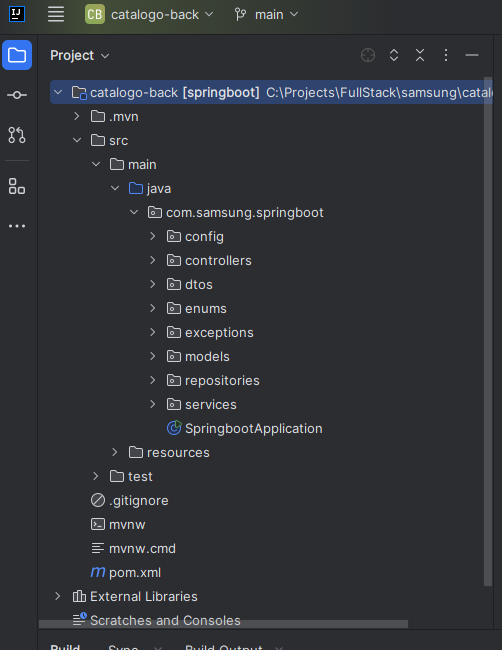
## Estrutura da Solução

Basicamente a solução foi dividida em dois projetos com as seguintes estruturas mostradas abaixo:

* Projeto Web (React + Vite) – Projeto estruturado em componentes o que nos ajuda a deixar o código mais manutenível, ou seja, mais fácil de fazer manutenção, já que temos as responsabilidades devidamente separadas. Isso também traz uma facilidade na compreensão do código, além da sua reutilização.



* Projeto Web API (Java Spring Boot) - Uma Web API (Application Programming Interface) é um conjunto de definições e protocolos que permite a comunicação entre software e serviços via HTTP.



## SGBD

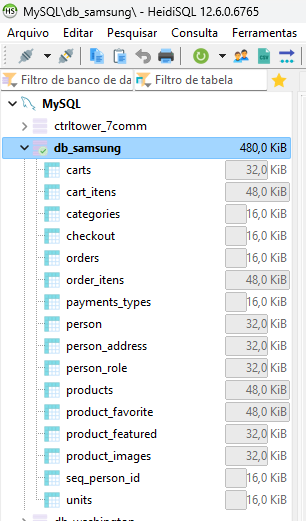
Usar um banco de dados MySQL oferece vantagens como a robustez na manipulação de dados complexos, suporte a consultas avançadas com SQL, consistência e integridade dos dados através de transações, além de ser amplamente adotado e suportado por ferramentas e frameworks diversos.



# Requisitos

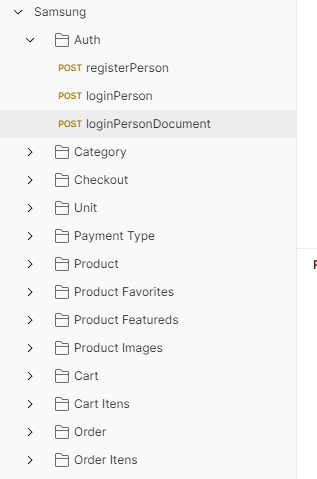
Para visualizar os requisitos de entrada consultar documento especifico enviado.

# Banco de Dados



# APIs

Para maiores detalhes consultar arquivo “Samsung.postman\_collection.json”



# Funcionamento

## Frontend

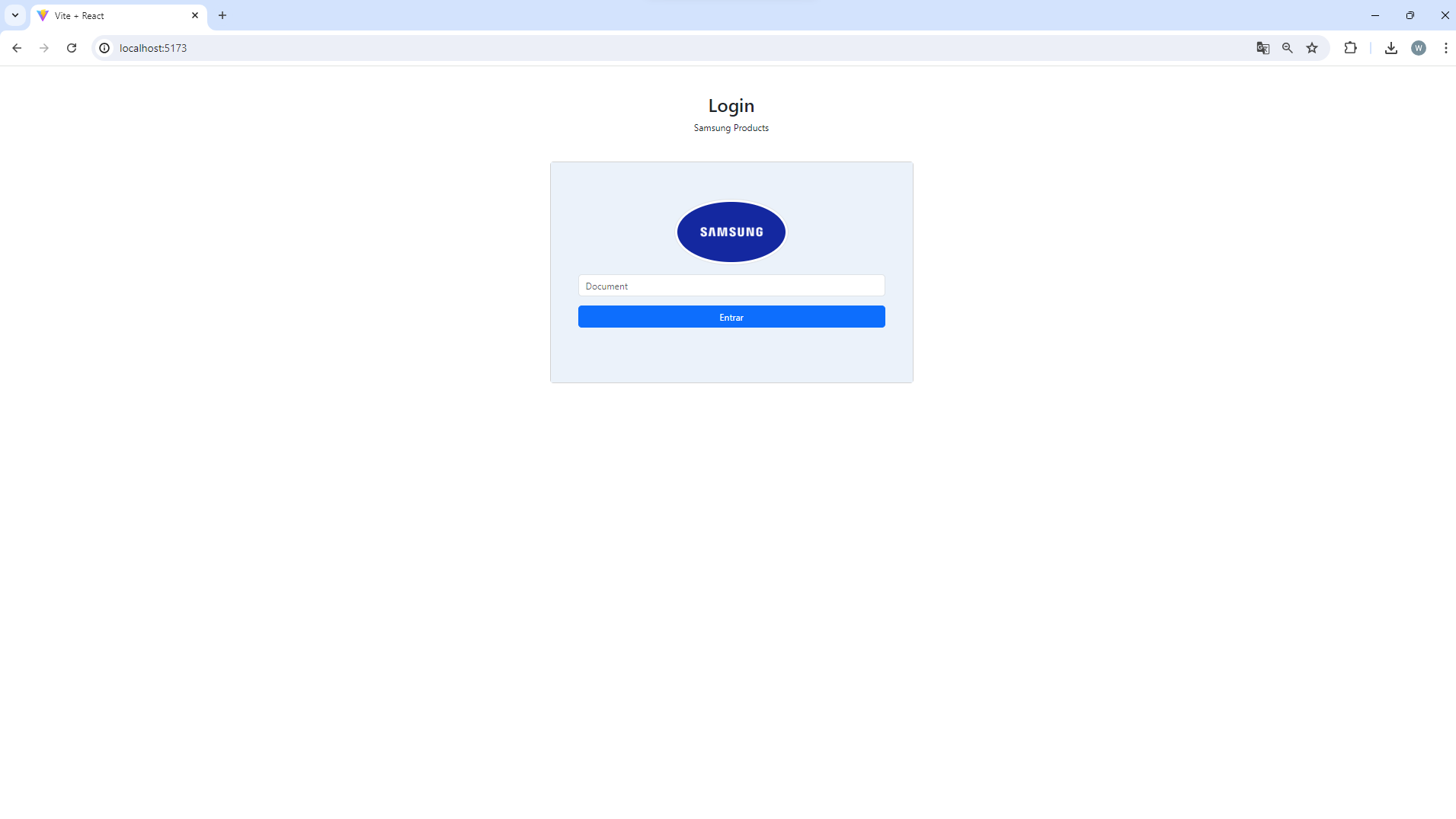
Explicar....

## Backend

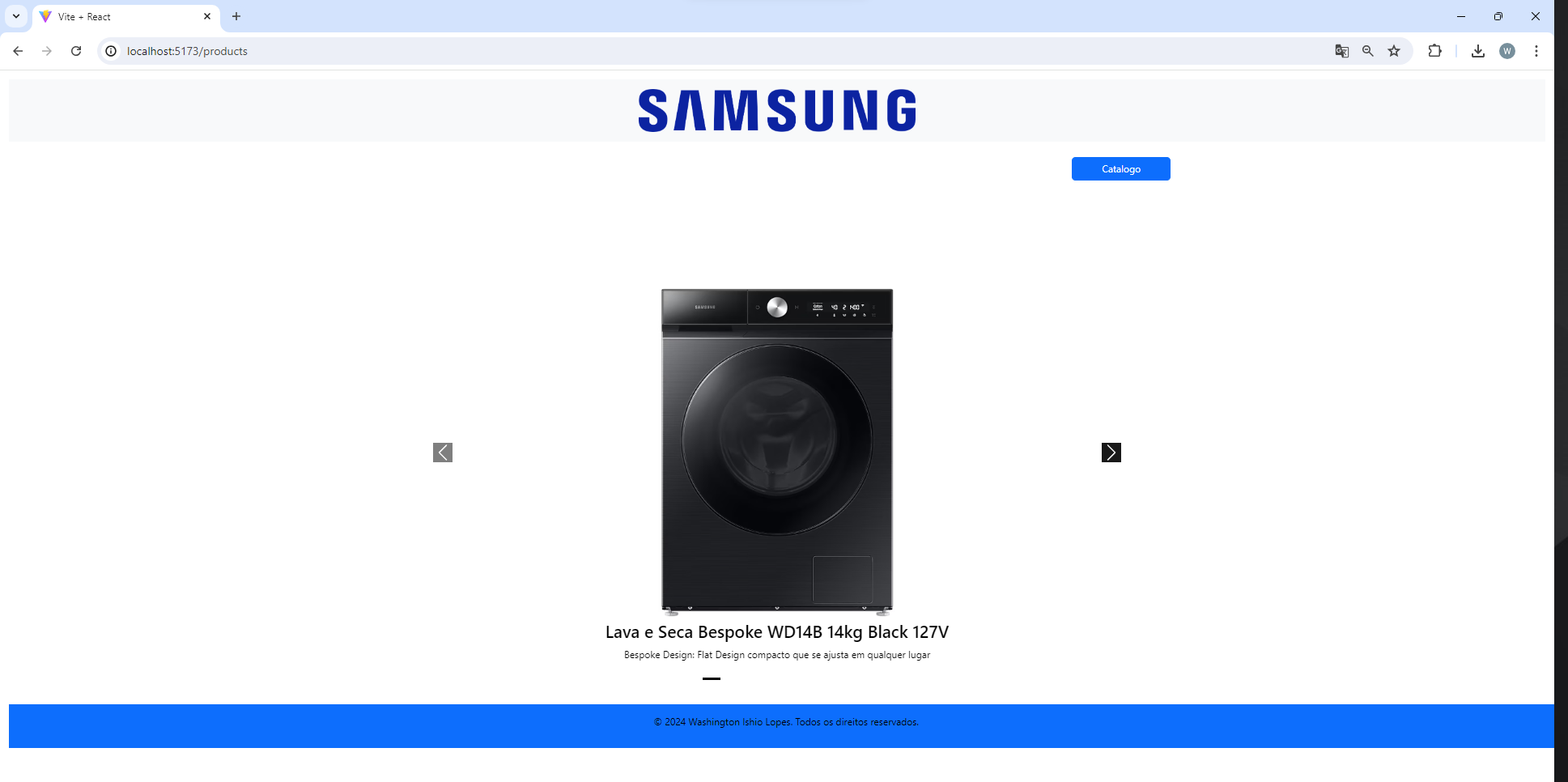
No backend foi implementado a autorização dos endpoints através Bearer Token. Também foi implementado CORS (\*.\*). Todos os endpoints foram implementados tentando seguir o padrão REST (não foi implementado HateOAS).

# Interface do Usuário

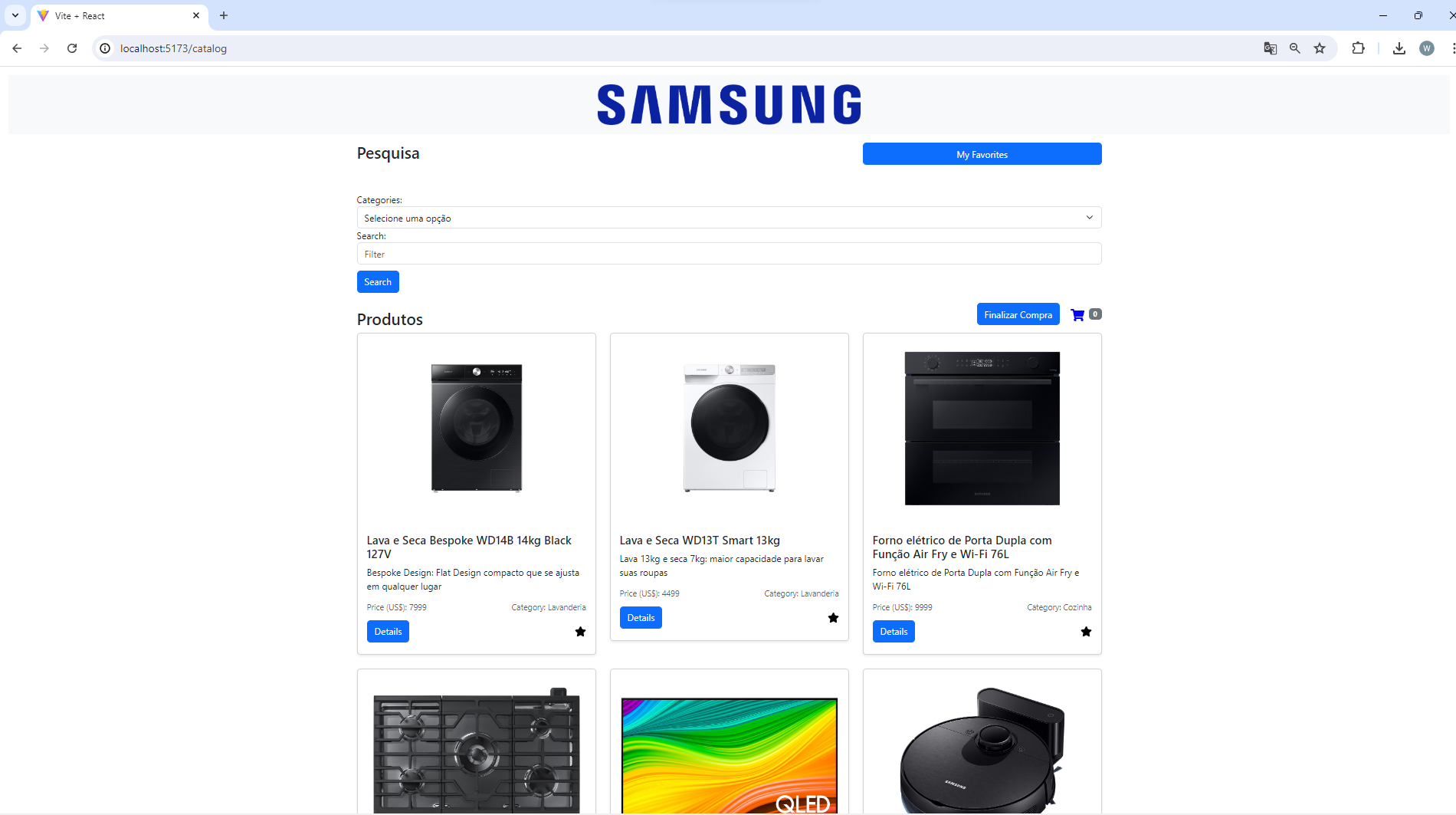
## Login com Documento



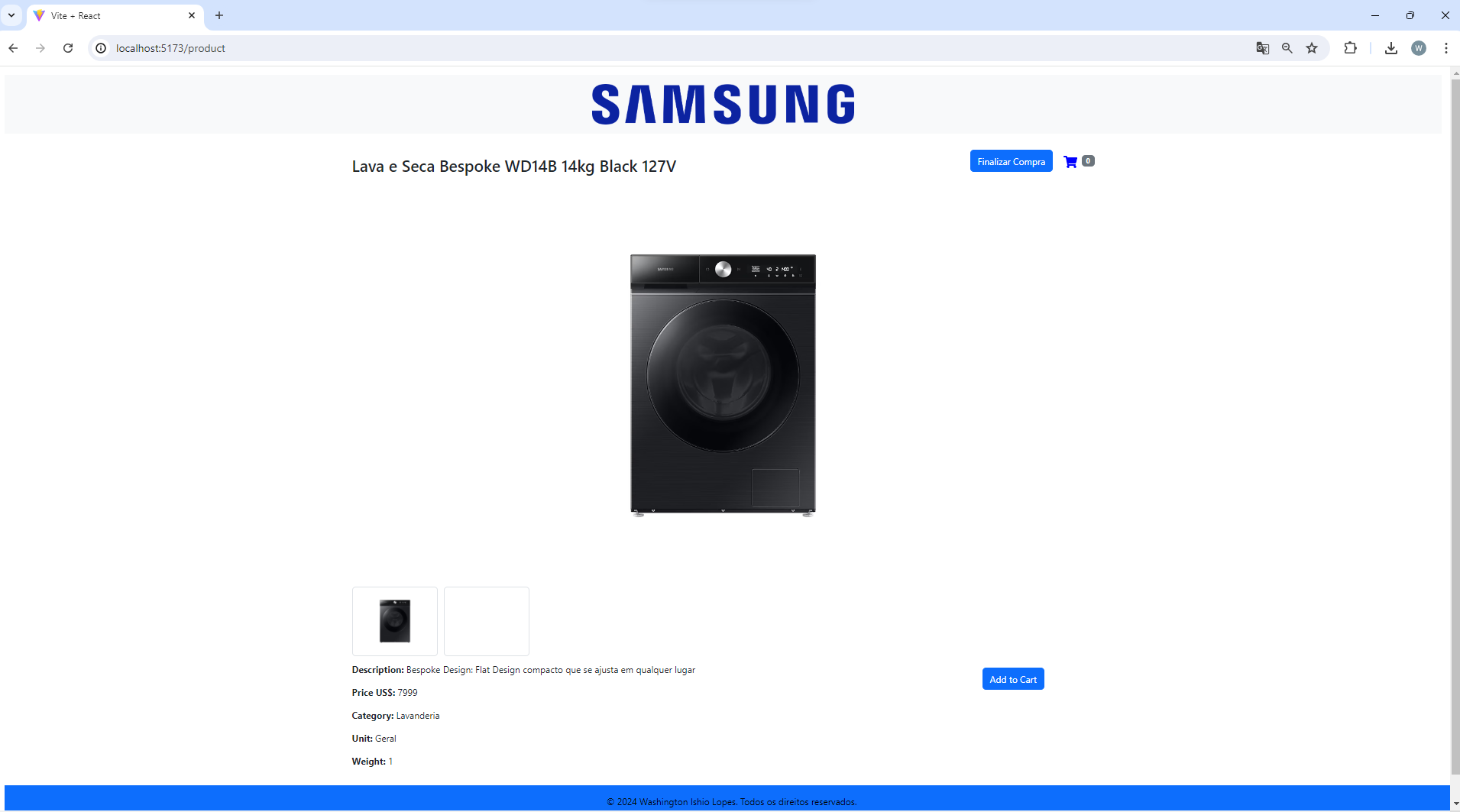
## Carrosel/Slider de Produtos



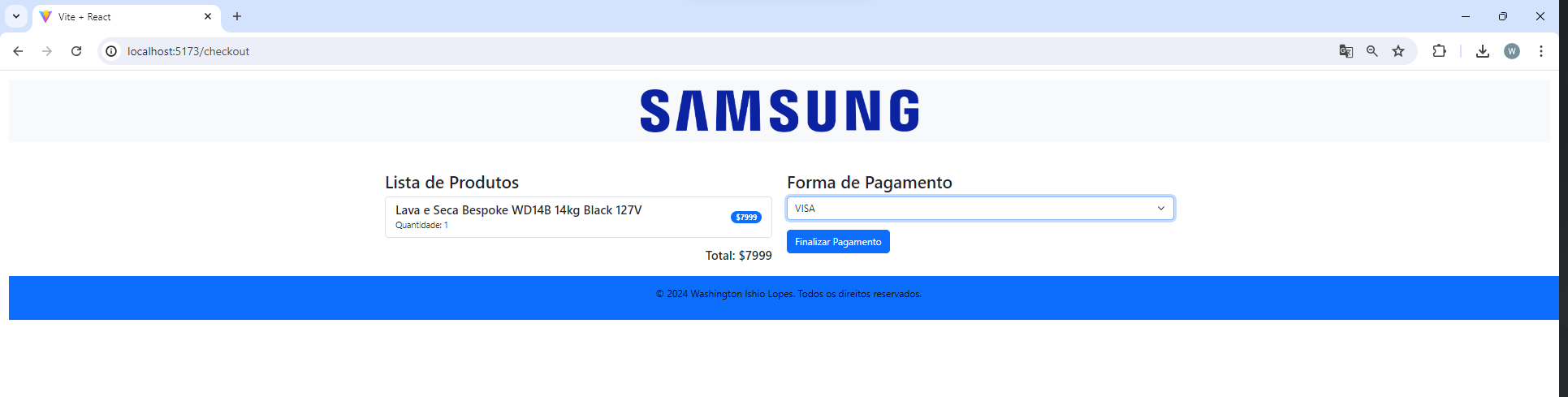
## Catalogo de Produtos



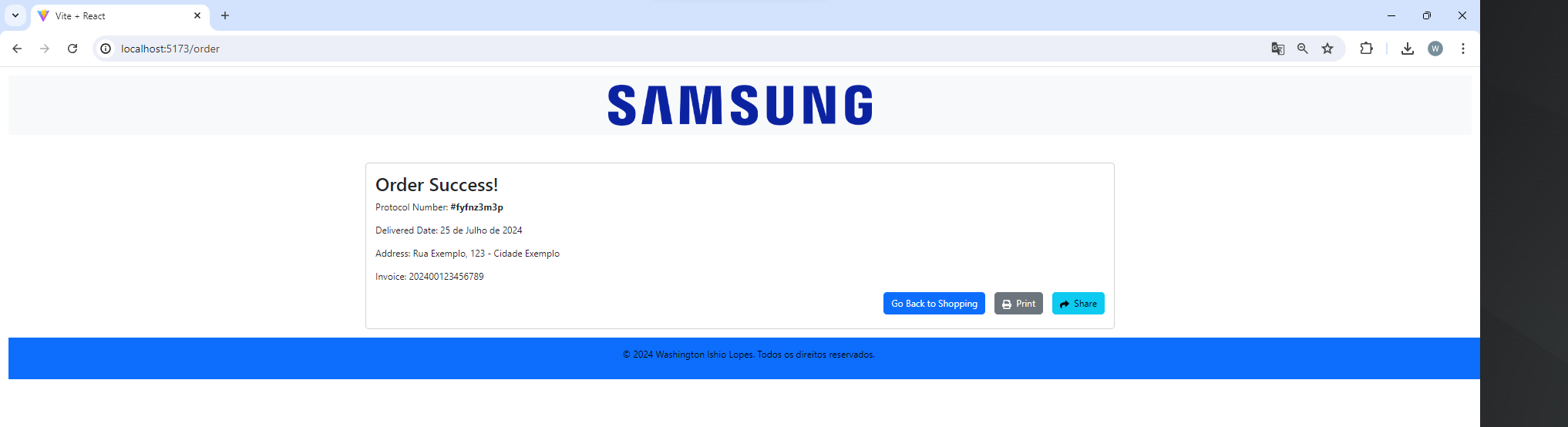
## Detalhes do Produto



## Checkout



## Comprovante



# Manutenção e Atualizações

Manutenção e Atualizações.....

# Pendencias

Segue abaixo as Pendencias:

1. Testar todos os Endpoints.
2. Funcionalidades do Meu Preferidos;
3. Consistência de quando a Lista de Imagens estiver vazia/null;

# Aprovação

**<Gestor>**

<gestor >

**<Gestor>**

<gestor técnico responsável>