**<<LOGO>>**

**ESPECIFICAÇÃO DE REQUISITOS DE SOFTWARE**

**E-commerce**

**Versão 1.0**

**setembro - 2024**

**Histórico de revisões**

|  |  |  |  |
| --- | --- | --- | --- |
| **Versão** | **Data** | **Autor** | **Descrição** |
| 1.0 | 09/09/2024 | Augusto | Iniciação do documento |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Sumário**

[**1 INTRODUÇÃO 5**](#_3gfapl2li6p7)

[1.1 Objetivo do documento 5](#_vs3ydg14iwza)

[1.2 Público alvo 5](#_s7maul1aqsg8)

[1.3 Convenções, termos e abreviações 5](#_645v3gffpnjn)

[1.3.1 Identificadores 5](#_nyyim6doudo1)

[1.3.2 Prioridade dos requisitos 5](#_pokr4bm59r48)

[1.3.3 Siglas e abreviações 6](#_dto2w7o7k7wl)

[1.4 Referências 6](#_r4q4omx7ivf4)

[**2 DESCRIÇÃO GERAL DO SISTEMA 7**](#_uq2puf1yjqb7)

[2.1 Visão geral 7](#_udroa4cx7141)

[2.2 Usuários 7](#_894hbqrdhq16)

[2.3 Benefícios 7](#_xv7t0om4zpga)

[2.4 Limitações e restrições 7](#_l3wyb8qcjqp9)

[**3 REQUISITOS 8**](#_woph2qrsdmhu)

[3.1 Requisitos Funcionais (RF) 8](#_tjrm8h8v63y7)

[3.1.1 <<Módulo 1>> 8](#_kf0xc3bqbik7)

[3.1.1.1 [RF001.01] <<Requisito>> 8](#_hbxuaolfc5uu)

[3.1.2 <<Módulo 2>> 8](#_armul0r72m8p)

[3.1.2.1 [RF002.01] <<Requisito>> 8](#_8ip3l2nsn2gh)

[3.2 Requisitos Não Funcionais (NF) 8](#_e5cbc74jozps)

[3.2.1 <<Módulo 1>> 8](#_cmrhv05hwd9y)

[3.2.1.1 [RNF001.01] <<Requisito>> 8](#_m0ytquf9e0wf)

[3.2.2 <<Módulo 2>> 8](#_vjyrei553htl)

[3.2.2.1 [RNF002.01] <<Requisito>> 8](#_u0mhnuc0zs4t)

[3.3 Regras de Negócio (RN) 8](#_ruzi8q3blavx)

[3.3.1 <<Módulo 1>> 8](#_qlta9ju5qnaw)

[3.3.1.1 [RN001.01] <<Requisito>> 8](#_7yoe9jinarq7)

[**4 RASTREABILIDADE 9**](#_1zxc4w4niejy)

[4.1 Rastreabilidade horizontal 9](#_3b6te046cwi7)

[**5 DIAGRAMAS E ILUSTRAÇÕES 9**](#_l3tljc76w6u7)

[5.1 Diagramas de Classe 10](#_fljglnrabrd5)

[5.2 Diagramas de Casos de Uso 10](#_fozt3k33owd7)

[**6 TESTES 11**](#_pq3v3ujvuvn3)

[6.1 Testes Unitários 11](#_dtoq68tf7afd)

[6.1.1 <<Módulo 1>> 11](#_xi9q31wtsqj6)

[6.1.1.1 [TU001.01] <<Teste>> 11](#_mvbavk2w77z5)

[6.2 Testes de Integração 11](#_1m74qn55enf9)

[6.2 Testes de Performance 11](#_wt5wnsver5aw)

[**7 PROTÓTIPOS DE TELAS 12**](#_xiuf0j8mkd49)

[7.1 Tela de Login 12](#_d217kn4i1t1o)

[7.2 Tela Inicial 12](#_hmpifxwqjyuv)

[7.3 Menus 12](#_9ghcafpnclvv)

[7.4 Tela de Cadastro de usuário 12](#_d7n5o0mjxrny)

# 1 INTRODUÇÃO

## 1.1 Objetivo do documento

Este documento tem o objetivo de detalhar os requisitos do software E-Commerce, fornecendo aos desenvolvedores as informações para o desenvolvimento, testes, homologação e implantação do sistema.

## 1.2 Público alvo

Este documento é destinado a equipe de desenvolvimento e aos usuários que participarão do projeto.

## 1.3 Convenções, termos e abreviações

### 1.3.1 Identificadores

Os requisitos serão identificados com um identificador único. A numeração inicia com o identificador **[RF001]** para o módulo acrescentado de **[.00]** para o requisito em si para requisitos funcionais e **[RNF001]** para o módulo acrescentado de **[.00]** para o requisito em si para requisitos não funcionais, a numeração será incrementada conforme surgirem novos requisitos.

Documentos anexos serão identificados com o identificador único **[DOC001]**, a numeração será incrementada conforme surgirem novos documentos.

Regras de negócios serão identificadas com o identificador único **[RN001]** para o módulo acrescentado de **[.00]** para a regra em si, a numeração será incrementada conforme surgirem novas regras.

Os testes serão identificados com o identificador único **[TU001]** para o módulo acrescentado de **[.00]** para o teste em si para testes unitários, **[TI001]**  para o módulo acrescentado de **[.00]** para o requisito em si para testes de integração e **[TP001]**  para o módulo acrescentado de **[.00]** para o requisito em si para teste de performance, a numeração será incrementada conforme surgirem novos testes.

### 1.3.2 Prioridade dos requisitos

Para estabelecer a prioridade dos requisitos, foram adotados as denominações **essenciais**, **importantes** e **desejáveis**, onde:

|  |  |
| --- | --- |
| **Essenciais** | Requisitos que são absolutamente necessários para o funcionamento do sistema. Sem esses requisitos, o software não pode operar ou cumprir sua finalidade principal. Eles são críticos e devem ser implementados. |
| **Importantes** | Requisitos que são muito desejáveis e que melhoram significativamente a funcionalidade do sistema, mas que não são críticos para o funcionamento básico. A ausência desses requisitos pode ser tolerada temporariamente, mas eles devem ser implementados assim que possível. |
| **Desejáveis** | Requisitos que podem ser implementados em futuras versões do software, caso haja tempo e recursos disponíveis. Elas não são essenciais para o lançamento inicial do produto. |

### 1.3.3 Siglas e abreviações

|  |  |  |
| --- | --- | --- |
| **Nº** | **Sigla** | **Definição** |
| 1 | ID | Identificador |
| 2 | SGBD | Sistema Gerenciador de Banco de Dados |

## 1.4 Referências

Destacar as referências que colaboraram com a criação do documento, exemplo, ata de entrevista com o cliente (participantes, ata gerada e onde encontrar o documento).

# 2 DESCRIÇÃO GERAL DO SISTEMA

## 2.1 Visão geral

Esse projeto será criado com o intuito de colocar em prática os conhecimentos adquiridos no decorrer dos estudos dos programadores.

Neste projeto será implementado tecnologias e metodologias utilizadas pelo mercado no período de criação do projeto.

## 2.2 Usuários

* Clientes:
  + Faixa etárias: Pessoas maiores de 18 anos.
  + Conhecimento em informática: básico.
  + Escolaridade: N/A.
* Usuário administrativo:
  + Faixa etárias: Pessoas maiores de 18 anos.
  + Conhecimento em informática: básico.
  + Escolaridade: Ensino médio completo.
* Usuário financeiro:
  + Faixa etárias: Pessoas maiores de 18 anos.
  + Conhecimento em informática: básico.
  + Escolaridade: Ensino superior completo.
* Usuário administrador:
  + Faixa etárias: Pessoas maiores de 18 anos.
  + Conhecimento em informática: avançado.
  + Escolaridade: Ensino técnico completo.

## 2.3 Benefícios

* Robustez: Com esse projeto esperamos construir um sistema com recursos que poderão atender do pequeno ao grande usuário.
* Segurança: Utilizaremos as tecnologias de segurança mais atuais para garantir a integridade dos dados dos clientes e usuários do sistema.
* Escalabilidade: O sistema estará preparado para crescer junto dos planos e projetos dos clientes garantindo uma entrega contínua dos serviços.
* Usabilidade: A usabilidade do sistema será de fácil entendimento tanto para os usuários quanto para os clientes, promovendo a satisfação de ambos.

## 2.4 Limitações e restrições

Nesse primeiro momento o frontend do sistema não terá um aplicativo móvel, apenas a versão para navegadores.

# 3 REQUISITOS

## 3.1 Requisitos Funcionais (RF)

### 3.1.1 Backend - Usuários

#### 3.1.1.1 [RF001.01] O sistema deve permitir o cadastro de usuários

**Prioridade:** Essencial.

**Descrição:** O sistema deve permitir o cadastros de usuários que acessarão o sistema com as informações conforme diagrama de classe 5.1.1.

#### 3.1.1.2 [RF001.02] O sistema deve permitir a alteração de usuários cadastrados

**Prioridade:** Essencial.

**Descrição:** O sistema deve permitir a alteração dos dados dos usuários cadastrados.

#### 3.1.1.3 [RF001.03] **O sistema deve ter perfis de usuários**

**Prioridade:** Essencial.

**Descrição:** O sistema deve ter perfis de usuários para o controle de acesso.

#### 3.1.1.4 [RF001.04] **O sistema deve autenticar usuários**

**Prioridade:** Essencial.

**Descrição:** O sistema deve autenticar a entrada de usuários através do login esenha.

#### 3.1.1.5 [RF001.05] **O sistema deve autorizar usuários**

**Prioridade:** Essencial.

**Descrição:** O sistema deve autorizar os usuários logados a acessarem as áreas baseadas no perfil.

### 3.1.2 <<Módulo>>

#### 3.1.2.1 <<Requisito>>

**Prioridade:** <<prioridade>>.

**Descrição:** <<descrição completa do requisito>>.

## 3.2 Requisitos Não Funcionais (NF)

### 3.2.1 Backend

#### 3.2.1.1 [RNF001.01] Linguagem de programação

**Prioridade:** Essencial.

**Descrição:** O backend do sistema deve ser elaborado utilizando a linguagem de programação Java na versão17.

#### 3.2.1.2 [RNF001.02] Gerenciador de dependências

**Prioridade:** Essencial.

**Descrição:** O backend do sistema deve ser elaborado utilizando o gerenciador de dependência Maven na versão 3.6.8.

#### 3.2.1.3 [RNF001.03] Framework

**Prioridade:** Essencial.

**Descrição:** O backend do sistema deve ser elaborado utilizando framework Spring na versão 3.3.3, utilizando dos seguintes projetos:

* Spring Web MVC: responsável por arquitetar as camadas de model e controller do projeto para criar as APIs Restful.
* Spring Validation: responsável por definir as validações dos dados.
* Spring HATEOAS: responsável por criar os hiperlinks de navegação.
* Spring Data JPA: responsável por abstrair a conexão e operações com o banco de dados.
* Spring AMQP: responsável por fazer a comunicação assíncrona utilizando um broker de mensageiria entre os micros serviços.
* Spring Cloud Netflix Eureka: responsável por criar um servidor de registro para gerenciar os micros serviços.
* Spring Cloud Gateway: responsável por criar a API Gateway.
* Spring Cloud Config: responsável por criar um servidor de configuração para os micros serviços.
* Spring Cloud Circuit Breaker Resilience4J: responsável por criar os métodos fallback para prevenir falhas.
* Spring Security: responsável por fazer a autenticação e autorização do projeto.

#### 3.2.1.4 [RNF001.04] Arquitetura

**Prioridade:** Essencial.

**Descrição:** O backend do sistema deve ser elaborado utilizando a arquitetura de micros serviços.

#### 3.2.1.5 [RNF001.05] Mensagens

**Prioridade:** Essencial.

**Descrição:** A troca de mensagens entre os micros serviços deve ser feita utilizando o broker RabbiMQ.

#### 3.2.1.6 [RNF001.06] Log agregation

**Prioridade:** Essencial.

**Descrição:** Os logs gerados pelos micros serviços devem ser centralizados utilizando o log agragation Elasticsearch.

#### 3.2.1.7 [RNF001.07] Banco de dados

**Prioridade:** Essencial.

**Descrição:** O banco de dados do projeto deve ser o PostgreSQL na versão 16.

### 3.2.2 <<Módulo 2>>

#### 3.2.2.1 [RNF002.01] <<Requisito>>

**Prioridade:** <<prioridade>>.

**Descrição:** <<descrição completa do requisito>>.

## 3.3 Regras de Negócio (RN)

### 3.3.1 Usuário

#### 3.3.1.1 [RN001.01] O nome do usuário deve ter entre 4 e 32 caracteres

**Prioridade:** Essencial.

**Descrição:** O sistema deve validar se o nome do usuário tem entre 4 e 32 caracteres, caso o nome esteja fora desses critérios o sistema não deve permitir o cadastro e deve retornar a mensagem para o usuário: “Informe um nome entre 4 e 32 caracteres”. Caso o nome esteja dentro dos critérios, o sistema deve permitir o cadastro.

#### 3.3.1.2 [RN001.02] O nome do usuário não deve conter apenas espaços em branco

**Prioridade:** Essencial.

**Descrição:** O sistema deve validar se o nome do usuário contém apenas espaços em branco, caso o nome esteja nessa condição o sistema não deve permitir o cadastro e deve retornar a mensagem para o usuário: “Nome do usuário inválido”. Caso o nome esteja fora dessa condição, o sistema deve permitir o cadastro.

#### 3.3.1.3 [RN001.03] O nome do usuário não deve conter apenas números

**Prioridade:** Essencial.

**Descrição:** O sistema deve validar se o nome do usuário contém apenas números, caso o nome esteja nessa condição o sistema não deve permitir o cadastro e deve retornar a mensagem para o usuário: “Nome do usuário inválido”. Caso o nome esteja fora dessa condição, o sistema deve permitir o cadastro.

#### 3.3.1.4 [RN001.04] O login do usuário deve ter entre 4 e 32 caracteres

**Prioridade:** Essencial.

**Descrição:** O sistema deve validar se o login do usuário tem entre 4 e 32 caracteres, caso o login esteja fora desses critérios o sistema não deve permitir o cadastro e deve retornar a mensagem para o usuário: “Informe um login entre 4 e 32 caracteres”. Caso o login esteja dentro dos critérios, o sistema deve permitir o cadastro.

#### 3.3.1.5 [RN001.05] O login do usuário não deve conter apenas espaços em branco

**Prioridade:** Essencial.

**Descrição:** O sistema deve validar se o login do usuário contém apenas espaços em branco, caso o login esteja nessa condição o sistema não deve permitir o cadastro e deve retornar a mensagem para o usuário: “Login inválido”. Caso o login esteja fora dessa condição, o sistema deve permitir o cadastro.

#### 3.3.1.6 [RN001.06] O login do usuário não deve conter apenas números

**Prioridade:** Essencial.

**Descrição:** O sistema deve validar se o login do usuário contém apenas números, caso o login esteja nessa condição o sistema não deve permitir o cadastro e deve retornar a mensagem para o usuário: “Login do usuário inválido”. Caso o login esteja fora dessa condição, o sistema deve permitir o cadastro.

#### 3.3.1.7 [RN001.07] O sistema não deve permitir duplicação de login de usuário

**Prioridade:** Essencial.

**Descrição:** O sistema deve validar se o login informado na hora do cadastro já existe no banco de dados, caso exista, o sistema não deve permitir o cadastro e deve retornar a mensagem para o usuário: “Login já cadastrado!”. Caso o login não exista, o sistema deve permitir o cadastro.

#### 3.3.1.8 [RN001.08] O sistema não deve permitir edição de login de usuário

**Prioridade:** Essencial.

**Descrição:** O sistema deve validar se o login foi editado na alteração dos dados cadastrais do usuário, caso o login tenha sido editado, o sistema não deve permitir a persistência dos dados e deve retornar a mensagem para o usuário: “Não é permitido alteração de login”. Caso o login não tenha sido alterado o sistema deve permitir a persistência dos dados.

#### 3.3.1.9 [RN001.09] A senha do usuário deve ter entre 4 e 32 caracteres

**Prioridade:** Essencial.

**Descrição:** O sistema deve validar se a senha do usuário tem entre 4 e 32 caracteres, caso a senha esteja fora desses critérios o sistema não deve permitir o cadastro e deve retornar a mensagem para o usuário: “A senha deve ter entre 4 e 32 caracteres”. Caso a senha esteja dentro dos critérios, o sistema deve permitir o cadastro.

#### 3.3.1.10 [RN001.10] A senha do usuário não deve conter apenas espaços em branco

**Prioridade:** Essencial.

**Descrição:** O sistema deve validar se a senha do usuário contém apenas espaços em branco, caso a senha esteja nessa condição o sistema não deve permitir o cadastro e deve retornar a mensagem para o usuário: “Senha inválida”. Caso a senha esteja fora dessa condição, o sistema deve permitir o cadastro.

# 4 RASTREABILIDADE

## 4.1 Rastreabilidade horizontal

Rastrear as regras de negócios sobre os requisitos funcionais utilizando matrizes.

Exemplo:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **RF001.01** | **RF002.01** | **…** | **RF N** |
| **RN001.01** | X |  | … | X |
| **RN002.01** |  | X | … |  |
| **…** | … | … | … | … |
| **RN N** | X | X | … |  |

# 5 DIAGRAMAS E ILUSTRAÇÕES

## 5.1 Diagramas de Classe

Representa a estrutura estática do sistema, mostrando as classes, seus atributos e métodos, e as relações entre elas. É útil para modelar a estrutura de dados e as classes que compõem o sistema.

**5.1.1 Usuário**



Figura 1: diagrama de classe Usuário

**5.1.1.1 Descrição**

* **Usuário:** Classe que representa os usuários que irão operar o sistema, é composta pelas classes UsuarioStatus e UsuarioTipo, possui os atributos:

|  |  |  |
| --- | --- | --- |
| **Atributos da classe Usuario** | | |
| **Atributo** | **Descrição** | **Tipo** |
| Id | Identificador da classe | UUID |
| Nome | Nome do usuário | String |
| Login | Login do usuário | String |
| Password | Senha do usuário | String |
| Telefone | Telefone do usuário | String |
| Email | E-mail do usuário | String |
| Status | Status do usuário | UsuarioStatus |
| Perfil | Perfil de acesso do usuário | UsuarioPerfil |

Tabela 1: atributos da classe Usuario

* **UsuarioStatus:** Classe do tipo ENUM que representa o status do usuário, compõe a classe Usuario, possui os atributos:

|  |  |  |
| --- | --- | --- |
| **Atributos da classe UsuarioStatus** | | |
| **Atributo** | **Descrição** | **Tipo** |
| ATIVO | Status ativo do usuário | ENUM |
| INATIVO | Status inativo do usuário | ENUM |

Tabela 2: atributos da classe UsuarioStatus

* **UsuarioPerfil:** Classe do tipo ENUM que representa o perfil de acesso do usuário, compõe a classe Usuario, possui os atributos:

|  |  |  |
| --- | --- | --- |
| **Atributos da classe UsuarioPerfil** | | |
| **Atributo** | **Descrição** | **Tipo** |
| ADMIN | Perfil administrador | ENUM |
| COMPRADOR | Perfil de comprador | ENUM |
| OPERADOR | Perfil de operador | ENUM |

Tabela 3: atributos da classe UsuarioPerfil

## 5.2 Diagramas de Casos de Uso

Descreve a funcionalidade do sistema do ponto de vista do usuário, mostrando as ações que os usuários podem realizar e os sistemas que os suportam. É ideal para capturar os requisitos funcionais do sistema.

# 6 TESTES

## 6.1 Testes Unitários

Descrever os testes unitários com base nos requisitos funcionais e regras de negócio.

### 6.1.1 <<Módulo 1>>

#### 6.1.1.1 [TU001.01] <<Teste>>

**Requisitos associados:** <<Requisitos atendidos no teste>>

**Descrição:** <<Descrição do teste>>

**Condições**: <<Condições para realização do teste>>

**Passos:** <<Passos para a execução dos teste>>

**Dados do teste:** <<Dados informados para o teste>>

**Resultado esperado:** <<Resultado esperado>>

**Status:** <<Situação do teste>>

**Realização do teste:** <<Data de realização>>

**Responsável:** <<Responsável pelo teste>>

## 6.2 Testes de Integração

Descrever os testes de integração com base nos requisitos funcionais e regras de negócio.

## 6.2 Testes de Performance

Descrever os testes de performance baseado em métricas estipuladas.

# 7 PROTÓTIPOS DE TELAS

## 7.1 Tela de Login

Adicionar imagem do protótipo.

## 7.2 Tela Inicial

Adicionar imagem do protótipo.

## 7.3 Menus

Adicionar imagem do protótipo.

## 7.4 Tela de Cadastro de usuário

Adicionar imagem do protótipo.