

# SAEP SIMULADO PRÁTICO II - DOCUMENTAÇÃO DO SISTEMA

## (DESENVOLVIMENTO DE SISTEMA DE GESTÃO DE ESTOQUE DE EQUIPAMENTOS ELETRÔNICOS).

### ENTREGA 1 - Lista de Requisitos Funcionais

Lista de Requisitos Funcionais (RFs) identificados a partir do desafio e das descrições das entregas:

- **RF001:** O sistema deve permitir que usuários do almoxarifado cadastrem produtos.
- **RF002:** O sistema deve permitir visualizar e gerenciar as entradas e saídas de produtos.
- **RF003:** O sistema deve implementar um mecanismo de controle de estoque mínimo.
- **RF004:** O sistema deve emitir alertas automáticos (visuais) quando o nível de estoque de um produto estiver abaixo do limite configurado.
- **RF005:** O sistema deve registrar um histórico completo das movimentações.
- **RF006:** O histórico de movimentações deve identificar o responsável (usuário), a data e o tipo de operação.
- **RF007:** O sistema deve possuir uma interface de autenticação de usuários (Login).
- **RF008:** O sistema deve exibir uma mensagem clara em caso de falha na autenticação.
- **RF009:** O sistema deve redirecionar o usuário novamente à tela de login após uma falha de autenticação.
- **RF010:** A interface principal do sistema deve exibir o nome do usuário logado.
- **RF011:** O sistema deve implementar uma função de logout com redirecionamento à tela de login.
- **RF012:** O sistema deve fornecer acesso às telas de "Cadastro de Produto" e "Gestão de Estoque" a partir da interface principal.
- **RF013:** A tela de "Cadastro de Produto" deve listar os produtos cadastrados em uma tabela.
- **RF014:** A tela de "Cadastro de Produto" deve possuir um campo de busca para filtrar produtos.
- **RF015:** O sistema deve permitir Inserir, Editar e Excluir produtos.
- **RF016:** O sistema deve validar campos obrigatórios no cadastro de produtos.
- **RF017:** A tela de "Gestão de Estoque" deve listar os produtos em ordem alfabética.
- **RF018:** O sistema deve permitir selecionar o produto e o tipo de movimentação (entrada ou saída).
- **RF019:** O sistema deve registrar a data da movimentação de estoque.

# ENTREGA 2 - Diagrama Entidade-Relacionamento (DER)

Descrição textual do modelo lógico (DER) implementado no arquivo database.sql:

O modelo é composto por três entidades principais:

## Entidade: Usuario

- id (PK, Inteiro, Autoincremento)
- username (Texto, Único, NOT NULL)
- password (Texto, NOT NULL)
- nome\_completo (Texto, NOT NULL)

## Entidade: Produto

- id (PK, Inteiro, Autoincremento)
- nome (Texto, NOT NULL)
- descricao (Texto)
- variacao (Texto, NOT NULL) - Ex: "16GB RAM, 512GB SSD" ou "128GB, Preto"
- quantidade\_atual (Inteiro, NOT NULL, Padrão 0)
- estoque\_minimo (Inteiro, NOT NULL, Padrão 5)

## Entidade: Movimentacao

- id (PK, Inteiro, Autoincremento)
- produto\_id (FK, Inteiro, NOT NULL) -> Refere-se a Produto(id)
- usuario\_id (FK, Inteiro, NOT NULL) -> Refere-se a Usuario(id)
- tipo (Texto, NOT NULL) - Restrito a 'Entrada' ou 'Saida'
- quantidade (Inteiro, NOT NULL)
- data\_hora (Datetime, NOT NULL)

## Relacionamentos:

- **1:N (Um-para-Muitos):** Um Usuario pode registrar múltiplas Movimentacao (1 Usuario -> N Movimentacao).
- **1:N (Um-para-Muitos):** Um Produto pode estar em múltiplas Movimentacao (1 Produto -> N Movimentacao).

## ENTREGA 3 - Script de Criação e População do Banco de Dados

Script SQL (para saep\_db.db) adaptado ao tema de Equipamentos Eletrônicos, com no mínimo 3 registros por tabela.

-- Limpa as tabelas se elas já existirem (para testes)

```
DROP TABLE IF EXISTS Movimentacao;
```

```
DROP TABLE IF EXISTS Produto;
```

```
DROP TABLE IF EXISTS Usuario;
```

-- Tabela de Usuários (Para Login e Rastreabilidade)

```
CREATE TABLE Usuario (  
    id INTEGER PRIMARY KEY AUTOINCREMENT,  
    username TEXT UNIQUE NOT NULL,  
    password TEXT NOT NULL,  
    nome_completo TEXT NOT NULL  
);
```

-- Tabela de Produtos

```
CREATE TABLE Produto (  
    id INTEGER PRIMARY KEY AUTOINCREMENT,  
    nome TEXT NOT NULL,  
    descricao TEXT,  
    variacao TEXT NOT NULL, -- Ex: "16GB RAM, 512GB SSD" ou "128GB, Preto"  
    quantidade_atual INTEGER NOT NULL DEFAULT 0,  
    estoque_minimo INTEGER NOT NULL DEFAULT 5  
);
```

-- Tabela de Histórico de Movimentações

```
CREATE TABLE Movimentacao (  
    id INTEGER PRIMARY KEY AUTOINCREMENT,  
    produto_id INTEGER NOT NULL,  
    usuario_id INTEGER NOT NULL,  
    tipo TEXT NOT NULL CHECK(tipo IN ('Entrada', 'Saida')),  
    quantidade INTEGER NOT NULL,  
    data_hora DATETIME NOT NULL,  
    FOREIGN KEY (produto_id) REFERENCES Produto(id),  
    FOREIGN KEY (usuario_id) REFERENCES Usuario(id)  
);
```

--- POPULAÇÃO DAS TABELAS (TEMA: ELETRÔNICOS) ---

-- Usuários (Senha '123' - Em produção, use HASH!)

```
INSERT INTO Usuario (username, password, nome_completo) VALUES  
( 'admin', '123', 'Administrador do Sistema'),  
( 'almox', '123', 'Funcionario Almoxarifado'),  
( 'gerente', '123', 'Gerente de Produção');
```

-- Produtos (Eletrônicos)

```
INSERT INTO Produto (nome, descricao, variacao, quantidade_atual, estoque_minimo)  
VALUES  
( 'Notebook', 'Notebook 15.6" Intel i7', '16GB RAM, 512GB SSD, Prata', 15, 5),  
( 'Smartphone', 'Smartphone 6.7" Tela OLED', '128GB, Preto, 5G', 40, 10),  
( 'Smart TV', 'Smart TV 55" 4K', 'LED, Wi-Fi, Tizen OS', 20, 8),  
( 'Monitor LED', 'Monitor Widescreen 24"', 'Full HD, HDMI/VGA, 75Hz', 30, 5);
```

# ENTREGA 8 - Descritivo de Casos de Teste de Software

## 8.2 Ferramentas e Ambientes de Teste

- **Ambiente de Execução:** Servidor Web local (localhost) executando a aplicação Flask em modo *debug*.
- **Framework/Linguagem:** Python 3.x com Flask.
- **Banco de Dados:** SQLite 3, acessado através do arquivo saep\_db.db.
- **Ferramenta de Teste (Manual):** Navegador Web (ex: Google Chrome, Mozilla Firefox) para testes funcionais e de interface.

## 8.1 Casos de Teste

ID	Requisito(s) Associado(s)	Descrição do Teste	Passos de Execução	Resultado Esperado
CT001	RF007 (Login com Sucesso)	Testar a autenticação de usuário com credenciais válidas.	<ol style="list-style-type: none"><li>1. Acessar a tela /login.</li><li>2. Inserir username: 'admin'.</li><li>3. Inserir password: '123'.</li><li>4. Clicar em "Entrar".</li></ol>	O usuário é redirecionado para a interface principal (/). A barra de navegação exibe a mensagem "Olá, Administrador do Sistema".

CT002	RF008, RF009 (Login com Falha)	Testar a autenticação de usuário com credenciais inválidas.	<ol style="list-style-type: none"> <li>1. Acessar a tela /login.</li> <li>2. Inserir username: 'admin'.</li> <li>3. Inserir password: 'senha_errada'.</li> <li>4. Clicar em "Entrar".</li> </ol>	O usuário é redirecionado de volta para /login. Uma mensagem de erro "Usuário ou senha inválidos." é exibida.
CT003	RF011 (Logout)	Testar a funcionalidade de logout do sistema.	<ol style="list-style-type: none"> <li>1. Realizar login (conforme CT001).</li> <li>2. Na barra de navegação, clicar em "Logout".</li> </ol>	O usuário é redirecionado para a tela /login. Uma mensagem informativa (ex: "Você saiu do sistema.") é exibida.
CT004	RF014, RF015 (Cadastro Produto)	Testar o cadastro de um novo produto.	<ol style="list-style-type: none"> <li>1. Fazer login e acessar a tela /produtos.</li> <li>2. Clicar em "Novo Produto".</li> <li>3. Preencher: Nome='Tablet', Variação='10 polegadas, 64GB'.</li> <li>4. Clicar em "Salvar".</li> </ol>	O produto "Tablet" aparece corretamente na listagem da tabela na página /produtos.

CT005	RF016 (Validação Cadastro)	Testar a validação de campos obrigatórios no cadastro de produto.	<p>1. Fazer login e acessar a tela /produtos.</p> <p>2. Clicar em "Novo Produto".</p> <p>3. Deixar o campo "Nome" vazio.</p> <p>4. Clicar em "Salvar".</p>	O cadastro não é realizado. Uma mensagem de erro (ex: "Nome e Variação são campos obrigatórios.") é exibida.
CT006	RF004, RF017, RF018 (Movimentação e Alerta de Estoque)	Testar o alerta visual de estoque mínimo após uma movimentação de saída.	<p>1. Fazer login.</p> <p>2. Acessar /produtos e garantir que "Notebook" tenha Est. Mínimo = 5 (conforme database.sql).</p> <p>3. Acessar /estoque (verificar que "Notebook" está listado em ordem alfabética).</p> <p>4. Registrar uma "Saída" de 11 unidades de "Notebook" (Estoque inicial 15 - 11 = 4).</p>	<p>O sistema registra a saída.</p> <p>5. Atualizar a tela /estoque.</p>

CT007	RF006 (Histórico)	Verificar se a movimentação do CT006 foi registrada corretamente no histórico.	1. Executar o CT006.  2. Acessar a tela /historico.	A tabela de histórico deve exibir uma nova linha contendo: Produto="Note book", Responsável="Administrador do Sistema" (ou quem logou), Tipo="Saida", Quantidade="1", e a Data/Hora corretas.
-------	----------------------	--------------------------------------------------------------------------------	-----------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## ENTREGA 9 - Lista de Requisitos de Infraestrutura

- **SGBD (Sistema de Gerenciamento de Banco de Dados):** SQLite, Versão 3.
- **Linguagem de Programação e Framework:** Python (versão 3.x) utilizando o framework Flask.
- **Sistema Operacional:** O sistema é agnóstico (multiplataforma), podendo ser executado em **Linux (ex: Ubuntu 22.04 LTS), Windows Server (2019 ou superior)** ou **macOS**, desde que possuam o interpretador Python 3 instalado.