

Luiz Felipe Neves dos Santos Siqueira

Programação orientada a objetos POO

Professor Carlos Veríssimo

Senac

GESTAO DE ESTOQUE Grupo 18

CHAPTER 2- CONTINUAÇÃO DA OUTRA ADO CASO DE USO

DAREMOS CONTINUIDADE A GESTAO DE ESTOQUE ATRAVES DO DIAGRAMA UML DE TODAS AS CLASSES

Caso de Uso 1 – Manter Funcionário

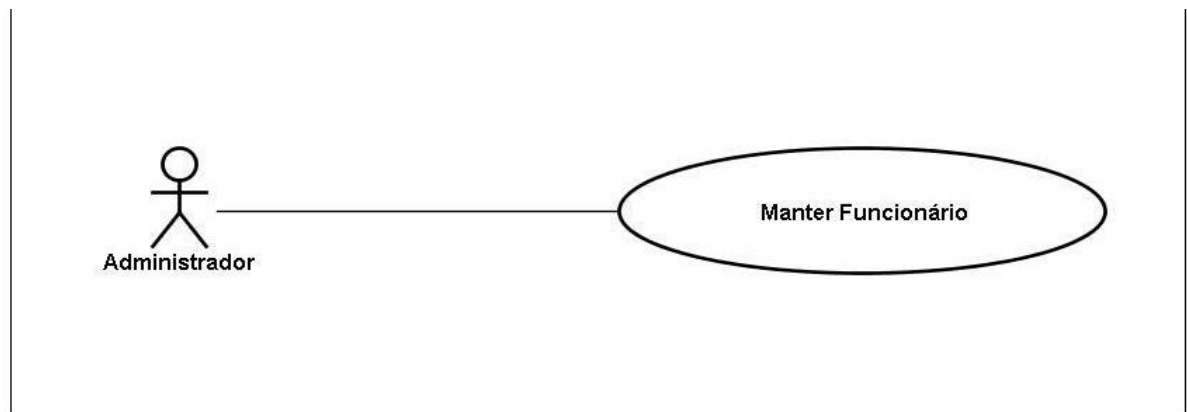


Diagrama de Caso de Uso Manter Funcionário

Funcionalidade/Objetivo	Inserir, alterar, excluir e pesquisar funcionário.
Ator	Administrador
Pré-Condição	O administrador deverá estar autenticado no sistema.

Cenário Principal	<p>1 – O sistema solicita os dados necessários para o cadastro do funcionário.</p> <p>2 – O administrador informa os dados de acordo com os campos a serem preenchidos.</p> <p>3 – O sistema solicita os dados para o cadastro da função.</p> <p>4 – O administrador informa os dados necessário.</p> <p>5 – O administrador seleciona a opção “Cadastrar”. 6 – o sistema emite a mensagem “Funcionário Cadastrado com Sucesso”.</p> <p>7 – O sistema cadastra o funcionário.</p>
Cenário Alternativo	<p>A1 - O administrador não informar os dados para o cadastro da função, o sistema informa que o funcionário não está cadastrado.</p> <p>A2 - O administrador poderá cancelar o processo durante o cadastro.</p>
Casos de Teste	<p>4.1- O sistema verifica se os campos foram preenchidos corretamente.</p> <p>4.2- O sistema não confirma o cadastro e emite uma mensagem de erro.</p> <p>4.3- O sistema cancela a operação.</p>

Quadro 1 – Manter Funcionário

Caso de Uso 2 – Manter Cliente

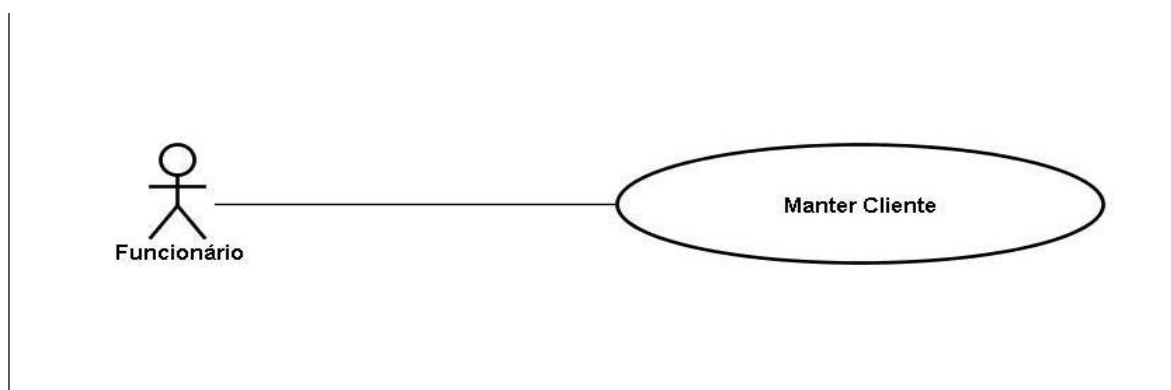
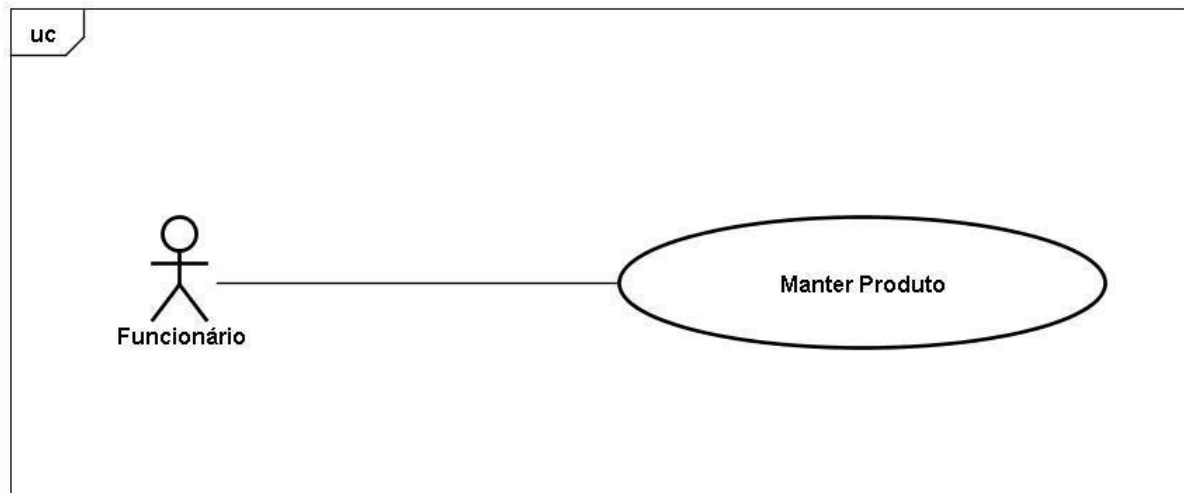


Diagrama de Caso de Uso Manter Cliente

Funcionalidade/Objetivo	Inserir, alterar, excluir e pesquisar cliente
Ator	Funcionário
Pré-Condição	O funcionário deverá estar autenticado no sistema.
Cenário Principal	<p>1 – O sistema solicita os dados necessários para o cadastro do cliente.</p> <p>2 – O funcionário informa os dados de acordo com os campos a serem preenchidos.</p> <p>3 – O sistema solicita os dados para o cadastro da função.</p> <p>4 – O funcionário informa os dados necessário. 5 – O Funcionário seleciona a opção “Cadastrar”.</p> <p>6 – O sistema emite a mensagem “Cliente Cadastrado com Sucesso”.</p> <p>7 – O sistema cadastra o cliente.</p>
Cenário Alternativo	<p>A1 – O funcionário não informar os dados para o cadastro da função, o sistema informa que o cliente não está cadastrado.</p> <p>A2 – O funcionário poderá cancelar o processo durante o cadastro.</p>
Casos Teste	<p>4.1 – O sistema verifica se os campos foram preenchidos corretamente.</p> <p>4.2 – O sistema não confirma o cadastro e emitir uma mensagem de erro.</p> <p>4.3 – O sistema cancela a operação.</p>

Quadro 2 – Manter Cliente

Caso de Uso 3 Manter Produto



3 Diagrama de Caso de Uso Manter Produto

Funcionalidade/Objetivo	Inserir, alterar, excluir e pesquisar produtos
Ator	Funcionário
Pré-Condição	O funcionário deverá estar autenticado no sistema
Cenário Principal	1 – O sistema solicita os dados necessários para o cadastro do produto. 2 – O funcionário informa os dados de acordo com os campos a serem preenchidos. 3 – O sistema solicita os dados para o cadastro da função. 4 – O funcionário informa os dados necessário. [A2] 5 – O funcionário seleciona a opção “Cadastrar”. 6 – O sistema emite a mensagem “Produto Cadastrado com Sucesso”. 7 – O sistema cadastra o produto.
Cenário Alternativo	A1 – Se o funcionário não informar os dados para o cadastro da função, o sistema informa que o produto não está cadastrado. A2 – O funcionário poderá cancelar o processo durante o cadastro.
Casos Teste	4.1- O sistema verifica se os campos foram preenchidos corretamente. 4.2 – O sistema não confirma o cadastro e emite uma mensagem de erro. 4.3 – O sistema cancela a operação.

Quadro 3 – Manter Produto

4 Movimentar Vendas

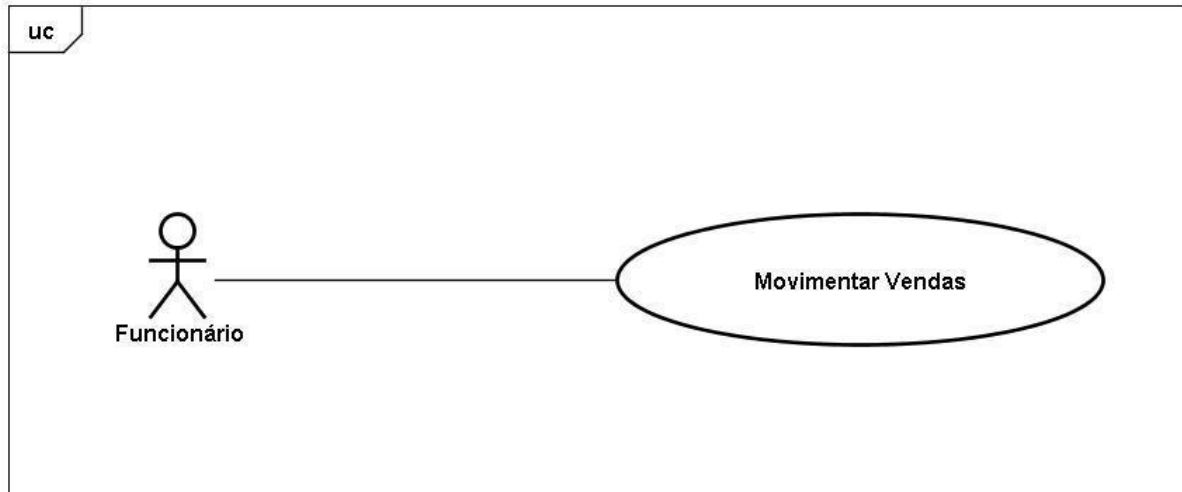
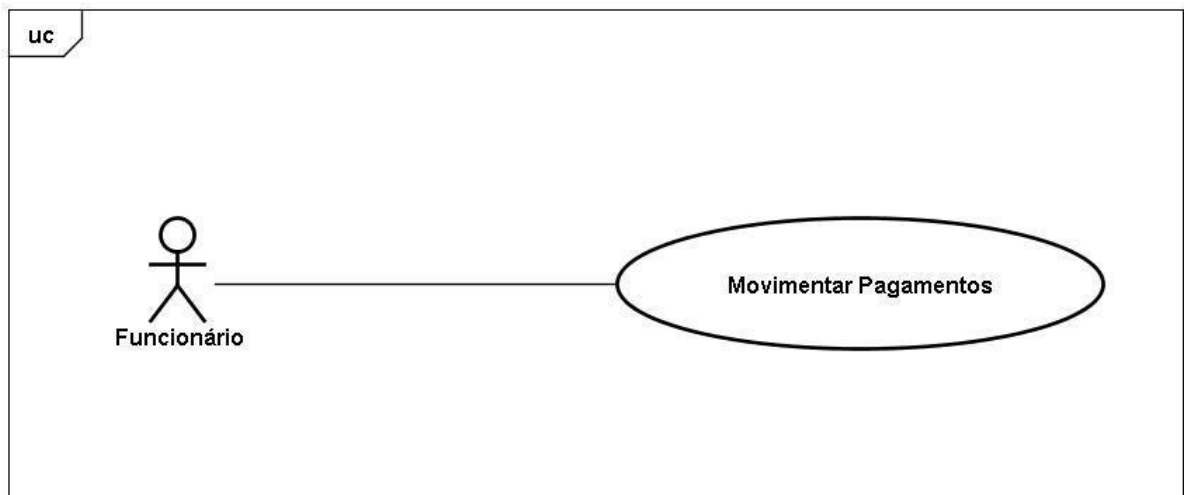


Diagrama de Caso de Uso Movimentar Vendas

Funcionalidade/Objetivo	Permite ao funcionário fornecer informações para a movimentação de vendas.
Ator	Funcionário
Pré-Condição	O funcionário deverá estar autenticado no sistema
Cenário Principal	1 – O sistema solicita os dados necessários para movimentar vendas. 2 – O funcionário informa os dados de acordo com os campos a serem preenchidos. 3 – O sistema solicita os dados para o cadastro da função. 4 – O funcionário informa os dados necessários. 5 – O funcionário seleciona a opção “Salvar”. 6 – O sistema emite a mensagem “Operação Realizada com Sucesso”.
Cenário Alternativo	A1 – O funcionário poderá cancelar o processo durante a movimentação.
Casos Teste	5.1 – O sistema verifica se os campos foram preenchidos corretamente. 5.2 – O sistema não confirma o cadastro e emite uma mensagem de erro. 5.3 – O sistema cancela a operação.

5 Caso de Uso 5 Movimentar Pagamentos



UC5 Diagrama de Caso de Uso Movimentar Pagamentos

Funcionalidade/Objetivo	Permite ao funcionário fornecer informações para a movimentação de pagamentos.
Ator	Funcionário
Pré-Condição	O funcionário deverá estar autenticado no sistema
Cenário Principal	1 – O sistema solicita os dados necessários para movimentar pagamento. 2 – O funcionário informa os dados de acordo com os campos a serem preenchidos. 3 – O sistema solicita os dados para o cadastro da função. 4 – O funcionário informa os dados necessários. 5 – O funcionário seleciona a opção “Salvar”. 6 – O sistema emite a mensagem “Operação Realizada com Sucesso”.
Cenário Alternativo	A1 – O funcionário poderá cancelar o processo durante a movimentação.
Casos Teste	5.1 – O sistema verifica se os campos foram preenchidos corretamente. 5.2 – O sistema não confirma o cadastro e emite uma mensagem de erro. 5.3 – O sistema cancela a operação.

Quadro 5 – Movimentar Pagamentos

6 Caso de Uso 6 Movimentar Recebimentos

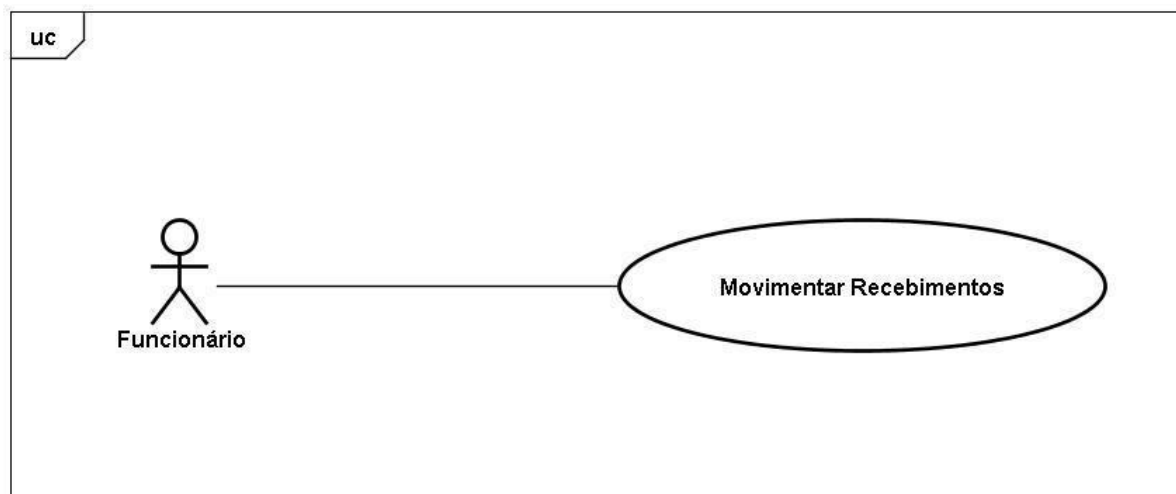
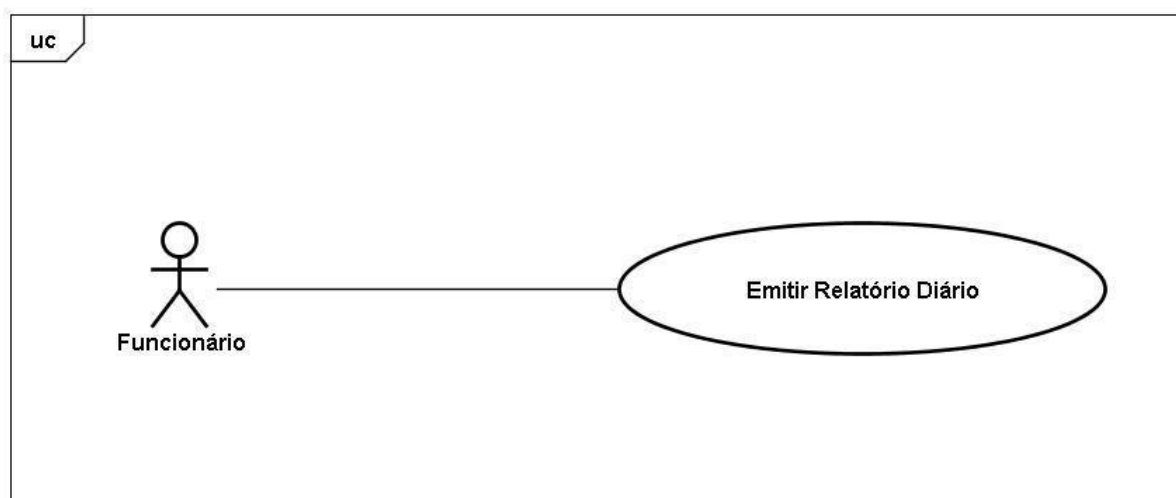


Diagrama de Caso de Uso Movimentar Recebimentos

Funcionalidade/Objetivo	Permite ao funcionário fornecer informações para a movimentação de recebimentos.
Ator	Funcionário
Pré-Condição	O funcionário deverá estar autenticado no sistema
Cenário Principal	<ol style="list-style-type: none">1 – O sistema solicita os dados necessários para movimentar recebimento.2 – O funcionário informa os dados de acordo com os campos a serem preenchidos.3 – O sistema solicita os dados para o cadastro da função.4 – O funcionário informa os dados necessários.5 – O funcionário seleciona a opção “Salvar”.6 – O sistema emite a mensagem “Operação Realizada com Sucesso”.
Cenário Alternativo	A1 – O funcionário poderá cancelar o processo durante a movimentação.

Casos Teste	<p>5.1 – O sistema verifica se os campos foram preenchidos corretamente.</p> <p>5.2 – O sistema não confirma o cadastro e emite uma mensagem de erro.</p> <p>5.3 – O sistema cancela a operação.</p>
-------------	--

7 Caso de Uso 7 Emitir Relatório Diário



Funcionalidade/Objetivo	Permite ao funcionário emitir relatório diário
Ator	Funcionário
Pré-Condição	O funcionário deverá estar autenticado no sistema
Cenário Principal	1 – O sistema disponibiliza os dados necessários para o relatório. 2 – O funcionário seleciona o botão “Relatório”. [A1] 3 – O funcionário seleciona o botão “Imprimir”. 4 – O sistema imprime o relatório com sucesso.
Cenário Alternativo	A1 – O funcionário poderá visualizar o relatório e não imprimir.
Casos Teste	4.1 – O funcionário cancela a operação.

Quadro 7 – Emitir Relatório Diário

Caso de Uso 8 Emitir Relatório Mensal

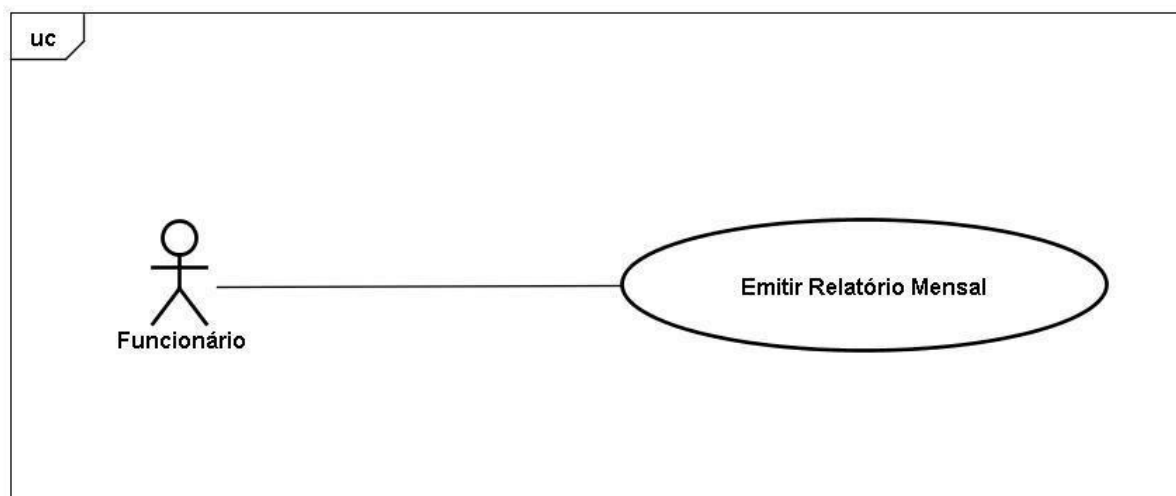


Figura 9 – UC8 Diagrama de Caso de Uso Emitir Relatório Mensal

Funcionalidade/Objetivo	Permite ao funcionário emitir relatório mensal
Ator	Funcionário
Pré-Condição	O funcionário deverá estar autenticado no sistema
Cenário Principal	1 – O sistema disponibiliza os dados necessários para o relatório. 2 – O funcionário seleciona o botão “Relatório”. [A1] 3 – O funcionário seleciona o botão “Imprimir”. 4 – O sistema imprime o relatório com sucesso.
Cenário Alternativo	A1 – O funcionário poderá visualizar o relatório e não imprimir.
Casos Teste	4.1 – O funcionário cancela a operação.

Quadro 8 – Emitir Relatório Mensal

Diagrama elaborado foi usado apenas um ator, o administrador, uma vez que o sistema será utilizado apenas pelo Adm, possuindo apenas o responsável por gerenciá-lo.

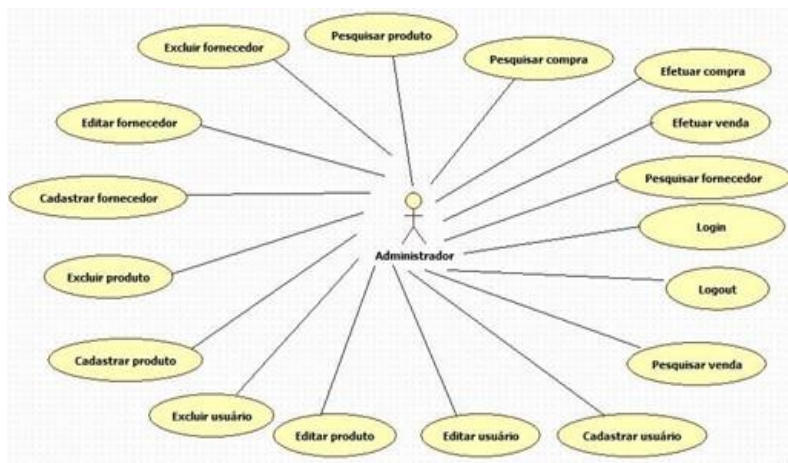
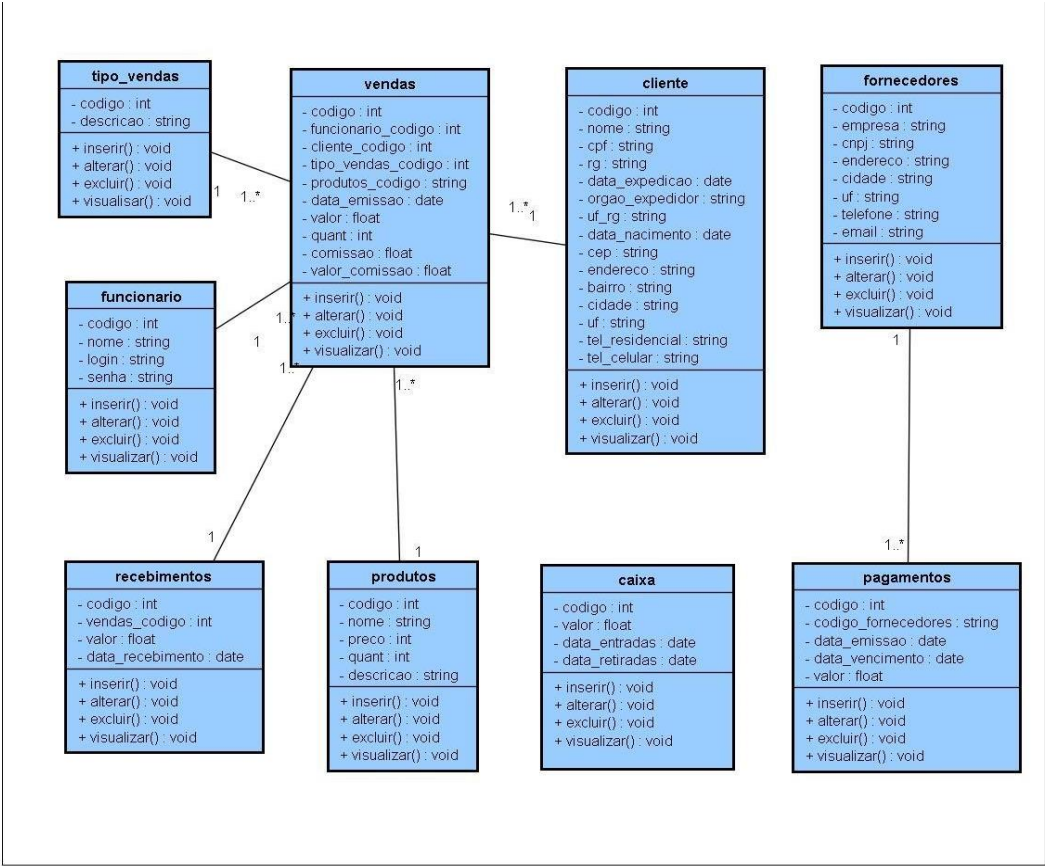


Diagrama de classe Mysql



(CODIGO INICIAL) JAVA ATRAVES DO DIAGRAMA DE CLASSE

```
import java.sql.SQLException;

public class GestaoEstoque {
    private Connection conexao;

    public GestaoEstoque(String url, String usuario, String senha) throws
SQLException {
        conexao = DriverManager.getConnection(url, usuario, senha);
    }

    public void adicionarProduto(String nome, int quantidade) throws
SQLException {
        String sql = "INSERT INTO estoque (nome, quantidade) VALUES (?,
?)";
        PreparedStatement stmt = conexao.prepareStatement(sql);
        stmt.setString(1, nome);
        stmt.setInt(2, quantidade);
        stmt.executeUpdate();
    }

    public void atualizarQuantidade(String nome, int quantidade) throws
SQLException {
        String sql = "UPDATE estoque SET quantidade = ? WHERE nome = ?";
        PreparedStatement stmt = conexao.prepareStatement(sql);
        stmt.setInt(1, quantidade);
        stmt.setString(2, nome);
        stmt.executeUpdate();
    }

    public int obterQuantidade(String nome) throws SQLException {
        String sql = "SELECT quantidade FROM estoque WHERE nome = ?";
```

```

        PreparedStatement stmt = conexao.prepareStatement(sql);
        stmt.setString(1, nome);
        ResultSet rs = stmt.executeQuery();
        if (rs.next()) {
            return rs.getInt("quantidade");
        } else {
            return 0;
        }
    }
}

```

Tabela para armazenar informações do produto

```

CREATE TABLE produtos (
    id INT AUTO_INCREMENT PRIMARY KEY,
    nome VARCHAR(255) NOT NULL,
    descricao TEXT,
    preco DECIMAL(10, 2) NOT NULL,
    quantidade INT NOT NULL
);

```

-- Tabela para armazenar informações de vendas

```

CREATE TABLE vendas (
    id INT AUTO_INCREMENT PRIMARY KEY,
    data_venda DATE NOT NULL,
    cliente_id INT,
    funcionario_id INT,
    caixa_id INT,
    FOREIGN KEY (cliente_id) REFERENCES clientes(id),
    FOREIGN KEY (funcionario_id) REFERENCES funcionarios(id),
    FOREIGN KEY (caixa_id) REFERENCES caixa(id)
);

```

-- Tabela para armazenar itens de venda

```

CREATE TABLE itens_venda (
    id INT AUTO_INCREMENT PRIMARY KEY,
    venda_id INT,
    produto_id INT,
    quantidade INT NOT NULL,
    preco_unitario DECIMAL(10, 2) NOT NULL,

```

```
FOREIGN KEY (venda_id) REFERENCES vendas(id),  
FOREIGN KEY (produto_id) REFERENCES produtos(id)  
);
```

-- Tabela para armazenar informações de clientes

```
CREATE TABLE clientes (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nome VARCHAR(255) NOT NULL,  
    endereco VARCHAR(255),  
    telefone VARCHAR(20)  
);
```

-- Tabela para armazenar informações de funcionários

```
CREATE TABLE funcionarios (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nome VARCHAR(255) NOT NULL,  
    cargo VARCHAR(255),  
    salario DECIMAL(10, 2)  
);
```

-- Tabela para armazenar informações de caixa

```
CREATE TABLE caixa (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    saldo DECIMAL(10, 2) NOT NULL  
);
```