



# DOCUMENTAÇÃO COMPLETA - ERP CORPORATIVO

## Portal do CEO - Sistema de Gestão Empresarial

---



### ÍNDICE

1. [INTRODUÇÃO](#)
  2. [MANUAL DO USUÁRIO](#)
    - 2.1 [Dashboard Principal](#)
    - 2.2 [Módulo de Usuários](#)
    - 2.3 [Módulo de Clientes](#)
    - 2.4 [Módulo de Fornecedores](#)
    - 2.5 [Módulo de Produtos](#)
    - 2.6 [Módulo de Estoque](#)
    - 2.7 [Módulo de Vendas](#)
    - 2.8 [Módulo de RH](#)
    - 2.9 [Módulo de Contratos](#)
    - 2.10 [Sistema de Solicitações](#)
  3. [GUIA DE TREINAMENTO](#)
    - 3.1 [Fluxos de Trabalho](#)
    - 3.2 [Estudos de Caso](#)
    - 3.3 [Exercícios Práticos](#)
    - 3.4 [Glossário](#)
  4. [DOCUMENTAÇÃO TÉCNICA](#)
    - 4.1 [Arquitetura do Sistema](#)
    - 4.2 [Tecnologias Utilizadas](#)
    - 4.3 [Estrutura de Pastas](#)
    - 4.4 [Endpoints REST](#)
    - 4.5 [Modelo de Banco de Dados](#)
    - 4.6 [Instalação e Configuração](#)
    - 4.7 [Deploy em Produção](#)
  5. [ANEXOS](#)
    - 5.1 [FAQ](#)
    - 5.2 [Checklist de Segurança](#)
    - 5.3 [Plano de Testes](#)
    - 5.4 [Registro de Mudanças](#)
- 

## 1. INTRODUÇÃO







### 1.1 Sobre o Sistema

O **Portal do CEO - ERP Corporativo** é um sistema completo de gestão empresarial desenvolvido com tecnologias modernas e robustas. Projetado especificamente para pequenas e médias empresas, oferece uma solução integrada para controle administrativo, financeiro, comercial e de recursos humanos.

### 1.2 Objetivos

- **Centralização:** Unificar todos os processos empresariais em uma única plataforma
- **Eficiência:** Automatizar tarefas repetitivas e otimizar fluxos de trabalho
- **Controle:** Fornecer visibilidade completa sobre todas as operações da empresa
- **Segurança:** Garantir proteção de dados com controle de acesso baseado em perfis
- **Escalabilidade:** Permitir crescimento e adaptação às necessidades futuras

## 1.3 Características Principais

-  **Interface Moderna:** Design responsivo e intuitivo
  -  **Modularidade:** Estrutura modular para fácil manutenção
  -  **Segurança Avançada:** Sistema de níveis de acesso hierárquico
  -  **Relatórios Dinâmicos:** Dashboards e relatórios em tempo real
  -  **Integração Completa:** Módulos totalmente integrados
  -  **Auditoria:** Rastreamento completo de ações dos usuários
- 

# 2. MANUAL DO USUÁRIO

## 2.1 Dashboard Principal

### 2.1.1 Visão Geral

O Dashboard é a página inicial do sistema, fornecendo uma visão consolidada dos principais indicadores da empresa.

### 2.1.2 Elementos do Dashboard

#### Indicadores Principais

1. **Total de Clientes:** Número total de clientes ativos
2. **Total de Vendas:** Valor total das vendas realizadas
3. **Produtos em Estoque:** Quantidade total de produtos disponíveis
4. **Últimas Vendas:** Lista das vendas mais recentes

#### Gráfico de Vendas

- Exibe vendas dos últimos 5 meses
- Atualização automática
- Visualização em gráfico de barras

### 2.1.3 Permissões Necessárias

- **Acesso:** Todos os usuários autenticados
- **Visualização Completa:** Níveis GERENTE ou superior
- **Dados Limitados:** Níveis OPERACIONAL e inferiores

### 2.1.4 Erros Comuns

**Problema:** Dashboard não carrega dados

- **Causa:** Problemas de conectividade com banco de dados
- **Solução:** Verificar conexão e contatar suporte técnico

**Problema:** Gráficos não aparecem

- **Causa:** JavaScript desabilitado no navegador
  - **Solução:** Habilitar JavaScript nas configurações do navegador
-

## 2.2 Módulo de Usuários

### 2.2.1 Funcionalidades Principais

#### Listar Usuários

1. Acesse **Usuários** → **Listar**
2. Visualize todos os usuários ativos do sistema
3. Use a barra de busca para filtrar por nome ou email
4. Clique em um usuário para ver detalhes

#### Cadastrar Novo Usuário

1. Acesse **Usuários** → **Cadastro**
2. Preencha os campos obrigatórios:
  - Nome completo
  - Email (será o login)
  - Senha
  - Confirmação de senha
  - Perfil de acesso
3. Selecione o cargo e departamento
4. Clique em **Salvar**

#### Editar Usuário

1. Na lista de usuários, clique em **Editar**
2. Modifique os campos necessários
3. **Atenção:** Não é possível alterar o próprio nível de acesso
4. Clique em **Salvar Alterações**

### 2.2.2 Níveis de Acesso

Nível	Descrição	Permissões
MASTER	Acesso total	Todas as funcionalidades
ADMIN	Administrador	Gerenciamento completo
GERENTE	Gerência	Supervisão e relatórios
COORDENADOR	Coordenação	Gestão de equipe
SUPERVISOR	Supervisão	Supervisão operacional
ANALISTA	Analítico	Análises e relatórios
OPERACIONAL	Operacional	Funções básicas
ESTAGIARIO	Estagiário	Acesso limitado

### 2.2.3 Permissões Necessárias

- **Visualizar:** COORDENADOR ou superior
- **Cadastrar:** ADMIN ou superior
- **Editar:** ADMIN ou superior
- **Excluir:** Apenas MASTER

## 2.2.4 Erros Comuns

**Problema:** "Email já cadastrado"

- **Solução:** Usar um email diferente ou verificar se o usuário já existe

**Problema:** "Senhas não conferem"

- **Solução:** Digitar a mesma senha nos campos "Senha" e "Confirmar Senha"

**Problema:** "Acesso negado"

- **Solução:** Verificar se possui permissão para a ação desejada
- 

## 2.3 Módulo de Clientes

### 2.3.1 Funcionalidades Principais

#### Listar Clientes

1. Acesse **Clientes** → **Lista**
2. Visualize estatísticas na parte superior:
  - Total de clientes
  - Clientes ativos/inativos
  - Pessoas físicas/jurídicas
3. Use filtros para buscar clientes específicos

#### Cadastrar Cliente

1. Acesse **Clientes** → **Cadastro**
2. Selecione o tipo: **Pessoa Física** ou **Pessoa Jurídica**
3. Preencha os dados básicos:
  - Nome/Razão Social
  - CPF/CNPJ
  - Email e telefones
4. Complete o endereço
5. Para PJ, preencha também:
  - Nome fantasia
  - Inscrição estadual/municipal
6. Clique em **Salvar**

#### Editar Cliente

1. Na lista, clique no cliente desejado
2. Clique em **Editar**
3. Modifique os campos necessários
4. Clique em **Salvar Alterações**

#### Ver Detalhes

1. Clique no nome do cliente na lista
2. Visualize informações completas
3. Acesse histórico de vendas
4. Veja contratos vinculados

### 2.3.2 Status de Clientes

- **Ativo:** Cliente com relacionamento comercial ativo
- **Inativo:** Cliente sem movimentação recente
- **Pendente:** Cliente em processo de aprovação
- **Bloqueado:** Cliente com restrições comerciais

### 2.3.3 Permissões Necessárias

- **Visualizar:** Todos os usuários
- **Cadastrar:** OPERACIONAL ou superior
- **Editar:** ANALISTA ou superior
- **Excluir:** ADMIN ou superior (exclusão lógica)

### 2.3.4 Erros Comuns

**Problema:** "CPF/CNPJ já cadastrado"

- **Solução:** Verificar se o cliente já existe no sistema

**Problema:** "CEP inválido"

- **Solução:** Verificar formato do CEP (00000-000)
- 

## 2.4 Módulo de Fornecedores

### 2.4.1 Funcionalidades Principais

#### Gerenciar Fornecedores

1. Acesse **Fornecedores** → **Listar**
2. Visualize todos os fornecedores cadastrados
3. Use filtros por status ou categoria

#### Cadastrar Fornecedor

1. Acesse **Fornecedores** → **Novo**
2. Preencha dados da empresa:
  - Razão social
  - CNPJ
  - Contatos
3. Defina categoria de produtos/serviços
4. Configure condições comerciais

#### Avaliações de Fornecedores

1. Acesse **Fornecedores** → **Avaliações**
2. Registre avaliações de desempenho
3. Critérios: qualidade, prazo, preço, atendimento
4. Gere relatórios de performance

#### Contratos com Fornecedores

1. Acesse **Fornecedores** → **Contratos**
2. Cadastre contratos de fornecimento
3. Defina prazos e condições
4. Acompanhe vencimentos

### 2.4.2 Permissões Necessárias

- **Visualizar:** OPERACIONAL ou superior
  - **Cadastrar:** ANALISTA ou superior
  - **Avaliar:** SUPERVISOR ou superior
  - **Contratos:** GERENTE ou superior
- 

## 2.5 Módulo de Produtos

### 2.5.1 Funcionalidades Principais

#### Cadastrar Produto

1. Acesse **Produtos** → **Novo**
2. Preencha informações básicas:
  - Nome do produto
  - Código EAN (opcional)
  - Código interno
  - Descrição
3. Defina preços e medidas:
  - Preço de venda
  - Unidade de medida
  - Dimensões e peso
4. Configure estoque:
  - Quantidade inicial
  - Estoque mínimo
5. Vincule categoria e fornecedor

#### Gerenciar Categorias

1. Acesse **Produtos** → **Categorias**
2. Crie categorias para organizar produtos
3. Defina hierarquia de categorias

### 2.5.2 Permissões Necessárias

- **Visualizar:** Todos os usuários
  - **Cadastrar:** ANALISTA ou superior
  - **Editar Preços:** SUPERVISOR ou superior
  - **Excluir:** GERENTE ou superior
- 

## 2.6 Módulo de Estoque

### 2.6.1 Funcionalidades Principais

#### Controle de Inventário

1. Acesse **Estoque** → **Inventário**
2. Visualize produtos em estoque
3. Monitore níveis críticos
4. Gere relatórios de posição

#### Movimentações de Estoque

##### Entrada de Produtos:

1. Acesse **Estoque** → **Entrada**
2. Selecione o produto
3. Informe quantidade e motivo
4. Registre fornecedor (se aplicável)
5. Confirme a entrada

#### **Saída de Produtos:**

1. Acesse **Estoque** → **Saída**
2. Selecione produto e quantidade
3. Informe motivo da saída
4. Confirme a operação

#### **Transferências**

1. Acesse **Estoque** → **Transferências**
2. Selecione origem e destino
3. Escolha produtos e quantidades
4. Registre a transferência

#### **Alertas de Estoque**

- Sistema gera alertas automáticos
- Produtos abaixo do estoque mínimo
- Produtos próximos ao vencimento
- Produtos sem movimentação

### **2.6.2 Tipos de Movimentação**

- **ENTRADA:** Compras, devoluções, ajustes positivos
- **SAIDA:** Vendas, perdas, ajustes negativos
- **TRANSFERENCIA:** Movimentação entre locais
- **AJUSTE:** Correções de inventário

### **2.6.3 Permissões Necessárias**

- **Visualizar:** OPERACIONAL ou superior
  - **Entrada:** ANALISTA ou superior
  - **Saída:** ANALISTA ou superior
  - **Ajustes:** SUPERVISOR ou superior
  - **Transferências:** GERENTE ou superior
- 

## **2.7 Módulo de Vendas**

### **2.7.1 Funcionalidades Principais**

#### **Registrar Venda**

1. Acesse **Vendas** → **Nova Venda**
2. Selecione o cliente
3. Adicione produtos:
  - Escolha produto
  - Defina quantidade
  - Confirme preço
4. Calcule totais automaticamente
5. Defina forma de pagamento

6. Finalize a venda

#### **Consultar Vendas**

1. Acesse **Vendas** → **Lista**
2. Filtre por período, cliente ou vendedor
3. Visualize detalhes de cada venda
4. Gere relatórios de performance

#### **2.7.2 Status de Vendas**

- **PENDENTE:** Venda registrada, aguardando pagamento
- **PAGA:** Pagamento confirmado
- **CANCELADA:** Venda cancelada
- **DEVOLVIDA:** Produtos devolvidos

#### **2.7.3 Permissões Necessárias**

- **Registrar:** OPERACIONAL ou superior
  - **Consultar Todas:** SUPERVISOR ou superior
  - **Cancelar:** GERENTE ou superior
  - **Relatórios:** ANALISTA ou superior
- 

## **2.8 Módulo de RH**

### **2.8.1 Funcionalidades Principais**

#### **Gestão de Colaboradores**

1. Acesse **RH** → **Colaboradores**
2. Cadastre novos funcionários
3. Gerencie informações pessoais e contratuais
4. Controle admissões e desligamentos

#### **Estrutura Organizacional**

##### **Departamentos:**

- Cadastre departamentos da empresa
- Defina hierarquia organizacional
- Associe colaboradores aos departamentos

##### **Cargos:**

- Crie cargos e funções
- Defina níveis hierárquicos
- Configure permissões por cargo

##### **Benefícios**

- Plano de saúde
- Vale transporte
- Vale refeição
- Outros benefícios customizáveis

##### **Controle de Ponto**

- Registro de entrada/saída
- Controle de horas extras



- Gestão de escalas
- Relatórios de frequência

#### **Folha de Pagamento**

- Cálculo automático de salários
- Gestão de descontos
- Geração de holerites
- Relatórios fiscais

#### **2.8.2 Permissões Necessárias**

- **Visualizar Equipe:** SUPERVISOR ou superior
  - **Cadastrar Colaborador:** GERENTE ou superior
  - **Folha de Pagamento:** GERENTE ou superior
  - **Relatórios RH:** COORDENADOR ou superior
- 

## **2.9 Módulo de Contratos**

### **2.9.1 Funcionalidades Principais**

#### **Gestão de Contratos**

1. Acesse **Contratos** → **Lista**
2. Visualize todos os contratos ativos
3. Monitore vencimentos
4. Acompanhe renovações

#### **Cadastrar Contrato**

1. Acesse **Contratos** → **Novo**
2. Defina tipo de contrato:
  - Fornecimento
  - Prestação de serviços
  - Locação
  - Outros
3. Preencha dados das partes envolvidas
4. Configure prazos e valores
5. Anexe documentos

#### **Acompanhamento**

- Alertas de vencimento
- Histórico de alterações
- Controle de renovações
- Relatórios de performance

### **2.9.2 Status de Contratos**

- **ATIVO:** Contrato vigente
- **VENCIDO:** Prazo expirado
- **CANCELADO:** Contrato cancelado
- **SUSPENSO:** Temporariamente suspenso

### **2.9.3 Permissões Necessárias**

- **Visualizar:** ANALISTA ou superior

- **Cadastrar:** SUPERVISOR ou superior
  - **Aprovar:** GERENTE ou superior
  - **Cancelar:** GERENTE ou superior
- 

## 2.10 Sistema de Solicitações

### 2.10.1 Funcionalidades Principais

#### Solicitar Acesso

1. Acesse **Solicitações** → **Nova Solicitação**
2. Preencha dados do solicitante
3. Justifique a necessidade de acesso
4. Selecione nível de acesso desejado
5. Envie para aprovação

#### Aprovar Solicitações

1. Acesse **Solicitações** → **Pendentes**
2. Analise cada solicitação
3. Verifique justificativas
4. Aprove ou rejeite com comentários

#### Acompanhar Status

1. Acesse **Solicitações** → **Minhas Solicitações**
2. Visualize status de suas solicitações
3. Receba notificações de mudanças

### 2.10.2 Status de Solicitações

- **PENDENTE:** Aguardando análise
- **APROVADA:** Solicitação aprovada
- **REJEITADA:** Solicitação negada
- **EM\_ANALISE:** Em processo de avaliação

### 2.10.3 Fluxo de Aprovação

1. **Solicitação** → Usuário cria solicitação
2. **Análise** → Gestor analisa necessidade
3. **Aprovação RH** → RH valida dados
4. **Aprovação TI** → TI aprova acesso técnico
5. **Criação** → Usuário é criado no sistema

### 2.10.4 Permissões Necessárias

- **Solicitar:** Todos os usuários
  - **Aprovar:** SUPERVISOR ou superior
  - **Gerenciar:** ADMIN ou superior
- 

## 3. GUIA DE TREINAMENTO

### 3.1 Fluxos de Trabalho

### **3.1.1 Fluxo de Cadastro de Novo Funcionário**

#### **Etapa 1: Admissão (RH)**

1. RH recebe documentação do novo funcionário
2. Cadastra colaborador no sistema
3. Define cargo e departamento
4. Registra dados contratuais

#### **Etapa 2: Solicitação de Acesso (Gestor)**

1. Gestor direto solicita acesso ao sistema
2. Justifica necessidade e nível de acesso
3. Sistema envia para aprovação

#### **Etapa 3: Aprovação (TI/Admin)**

1. TI analisa solicitação
2. Verifica conformidade com políticas
3. Cria usuário no sistema
4. Envia credenciais por email

#### **Etapa 4: Primeiro Acesso (Funcionário)**

1. Funcionário recebe credenciais
2. Realiza primeiro login
3. Altera senha obrigatoriamente
4. Completa perfil pessoal

### **3.1.2 Fluxo de Venda Completa**

#### **Etapa 1: Cadastro do Cliente**

1. Vendedor cadastra novo cliente
2. Coleta dados completos
3. Valida informações

#### **Etapa 2: Criação da Venda**

1. Seleciona cliente
2. Adiciona produtos ao pedido
3. Negocia preços e condições
4. Confirma disponibilidade em estoque

#### **Etapa 3: Finalização**

1. Define forma de pagamento
2. Gera documento fiscal
3. Baixa estoque automaticamente
4. Envia confirmação ao cliente

#### **Etapa 4: Pós-Venda**

1. Acompanha entrega
2. Confirma satisfação do cliente
3. Registra feedback
4. Programa follow-up

### 3.1.3 Fluxo de Compras

#### Etapa 1: Identificação da Necessidade

1. Sistema gera alerta de estoque baixo
2. Gestor analisa necessidade
3. Aprova solicitação de compra

#### Etapa 2: Cotação

1. Solicita cotações de fornecedores
2. Compara preços e condições
3. Seleciona melhor proposta

#### Etapa 3: Pedido de Compra

1. Gera pedido de compra
2. Envia para fornecedor
3. Acompanha prazo de entrega

#### Etapa 4: Recebimento

1. Confere produtos recebidos
2. Registra entrada no estoque
3. Libera pagamento
4. Avalia fornecedor

## 3.2 Estudos de Caso

### 3.2.1 Caso 1: Empresa de Varejo

**Situação:** Loja de roupas com 50 funcionários

#### Desafios:

- Controle de estoque sazonal
- Gestão de vendedores comissionados
- Múltiplas formas de pagamento

#### Solução Implementada:

1. Configuração de categorias sazonais
2. Sistema de comissões automático
3. Integração com meios de pagamento
4. Relatórios de performance por vendedor

#### Resultados:

- 30% redução no tempo de fechamento
- 25% aumento na precisão do estoque
- 40% melhoria na gestão de comissões

### 3.2.2 Caso 2: Empresa de Serviços

**Situação:** Consultoria com 30 colaboradores

#### Desafios:

- Controle de projetos
- Gestão de contratos
- Faturamento por horas

**Solução Implementada:**

1. Módulo de projetos customizado
2. Controle de contratos por cliente
3. Sistema de apontamento de horas
4. Faturamento automático

**Resultados:**

- 50% redução no tempo de faturamento
- 35% melhoria no controle de projetos
- 20% aumento na margem de lucro

### 3.2.3 Caso 3: Indústria Pequeno Porte

**Situação:** Fábrica de móveis com 80 funcionários

**Desafios:**

- Controle de matéria-prima
- Gestão de produção
- Controle de qualidade

**Solução Implementada:**

1. Controle rigoroso de estoque
2. Rastreabilidade de produtos
3. Sistema de qualidade integrado
4. Relatórios de produtividade

**Resultados:**

- 45% redução no desperdício
- 30% melhoria na qualidade
- 25% aumento na produtividade

## 3.3 Exercícios Práticos

### 3.3.1 Exercício 1: Cadastro Completo

**Objetivo:** Praticar cadastro de dados básicos

**Instruções:**

1. Cadastre 3 clientes (2 PF e 1 PJ)
2. Cadastre 5 produtos de categorias diferentes
3. Cadastre 2 fornecedores
4. Registre entrada de estoque para todos os produtos

**Critérios de Avaliação:**

- Completude dos dados
- Consistência das informações
- Uso correto das funcionalidades

### 3.3.2 Exercício 2: Processo de Venda

**Objetivo:** Simular processo completo de venda

**Instruções:**

1. Selecione um cliente cadastrado
2. Crie uma venda com 3 produtos diferentes
3. Aplique desconto de 10%
4. Finalize com pagamento à vista
5. Verifique baixa automática do estoque

**CrITÉrios de Avaliação:**

- Correção dos cálculos
- Atualização do estoque
- Geração de documentos

### 3.3.3 Exercício 3: Gestão de Usuários

**Objetivo:** Praticar gestão de acessos

**Instruções:**

1. Crie solicitação de acesso para novo usuário
2. Aprove a solicitação
3. Cadastre o usuário com nível OPERACIONAL
4. Teste login com as credenciais criadas
5. Altere nível para ANALISTA

**CrITÉrios de Avaliação:**

- Fluxo de aprovação correto
- Configuração adequada de permissões
- Funcionamento do login

## 3.4 Glossário

### Termos Técnicos

**API:** Interface de Programação de Aplicações - permite integração entre sistemas

**Backend:** Parte do sistema que processa dados e regras de negócio

**Dashboard:** Pannel principal com indicadores e métricas importantes

**Frontend:** Interface visual que o usuário interage

**CRUD:** Create, Read, Update, Delete - operações básicas de dados

**JWT:** JSON Web Token - método de autenticação segura

**ORM:** Object-Relational Mapping - mapeamento objeto-relacional

**REST:** Representational State Transfer - arquitetura para APIs

### Termos de Negócio

- B2B:** Business to Business - negócios entre empresas
- B2C:** Business to Consumer - negócios com consumidor final
- CRM:** Customer Relationship Management - gestão de relacionamento com cliente
- ERP:** Enterprise Resource Planning - planejamento de recursos empresariais
- KPI:** Key Performance Indicator - indicador chave de performance
- ROI:** Return on Investment - retorno sobre investimento
- SLA:** Service Level Agreement - acordo de nível de serviço
- Workflow:** Fluxo de trabalho automatizado

**Termos do Sistema**

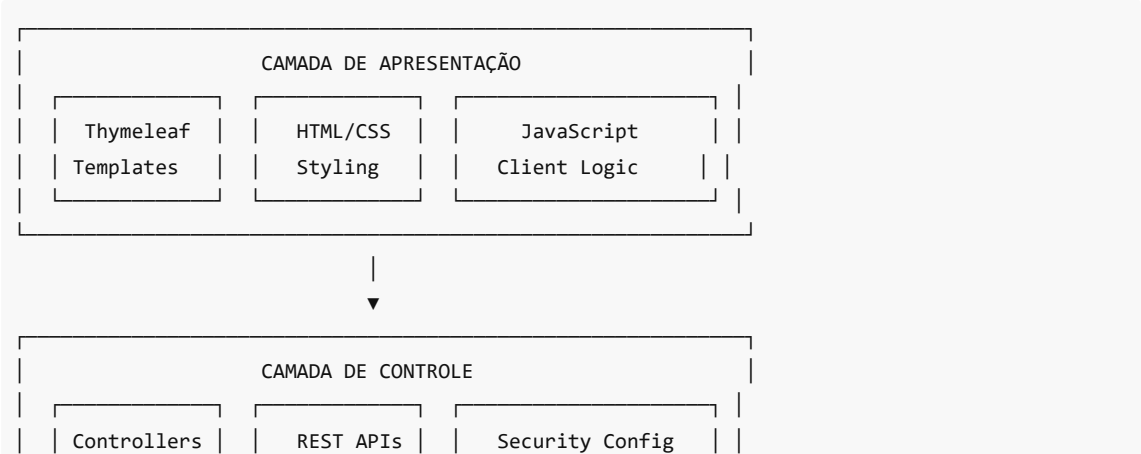
- Colaborador:** Funcionário da empresa cadastrado no RH
- Usuário:** Pessoa com acesso ao sistema (login e senha)
- Perfil:** Conjunto de permissões agrupadas
- Nível de Acesso:** Hierarquia de autoridade no sistema
- Solicitação:** Pedido de acesso ou alteração no sistema
- Auditoria:** Registro de ações realizadas no sistema
- Exclusão Lógica:** Marcação como excluído sem remoção física
- Estoque Mínimo:** Quantidade mínima para gerar alerta

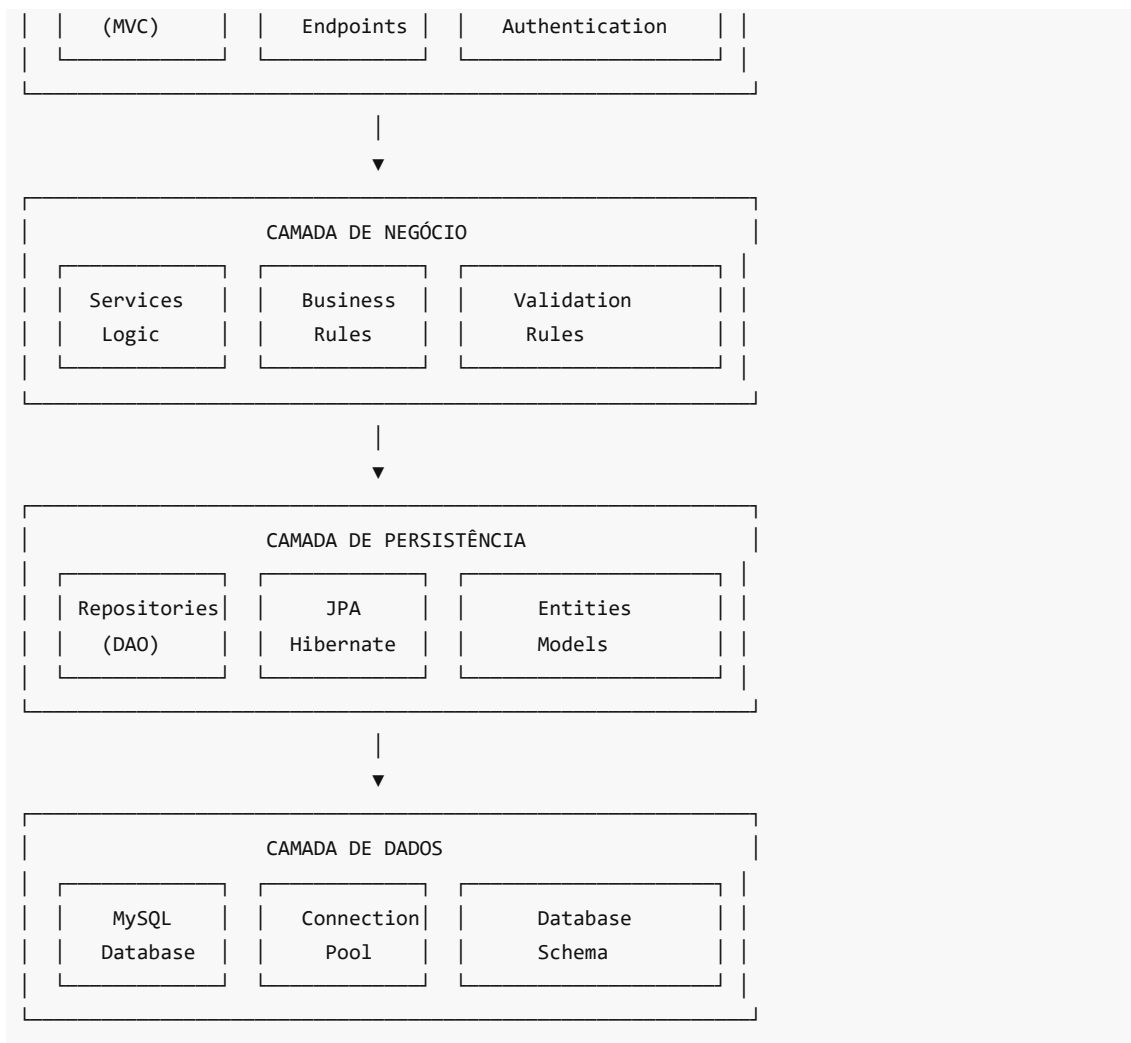
# 4. DOCUMENTAÇÃO TÉCNICA

## 4.1 Arquitetura do Sistema

### 4.1.1 Visão Geral da Arquitetura

O sistema segue uma arquitetura em camadas (Layered Architecture) baseada no padrão MVC (Model-View-Controller), proporcionando separação clara de responsabilidades e facilidade de manutenção.





#### 4.1.2 Padrões Arquiteturais Utilizados

##### MVC (Model-View-Controller)

- **Model:** Entidades JPA representando dados de negócio
- **View:** Templates Thymeleaf para renderização
- **Controller:** Classes Spring MVC para controle de fluxo

##### Repository Pattern

- Abstração da camada de acesso a dados
- Interfaces Spring Data JPA
- Queries customizadas quando necessário

##### Service Layer Pattern

- Lógica de negócio centralizada
- Transações gerenciadas pelo Spring
- Validações e regras de negócio

##### DTO (Data Transfer Object)

- Transferência de dados entre camadas
- Redução de acoplamento
- Controle de exposição de dados



### 4.1.3 Componentes Principais

#### Configurações (Config)

- SecurityConfig : Configuração de segurança e autenticação
- DateConfig : Formatação e conversão de datas
- MultipartConfig : Upload de arquivos
- GlobalControllerAdvice : Tratamento global de exceções

#### Segurança (Security)

- UsuarioDetailsService : Carregamento de dados do usuário
- CustomAuthenticationFailureHandler : Tratamento de falhas de login
- Integração com Spring Security

#### Exceções (Exception)

- GlobalExceptionHandler : Tratamento centralizado de exceções
- Exceções customizadas para regras de negócio
- Páginas de erro personalizadas

## 4.2 Tecnologias Utilizadas

### 4.2.1 Backend

#### Spring Boot 3.5.3

- Framework principal para desenvolvimento
- Auto-configuração e convenções
- Embedded server (Tomcat)
- Produtividade e rapidez no desenvolvimento

#### Spring Security

- Autenticação e autorização
- Proteção contra ataques comuns
- Controle de acesso baseado em roles
- Criptografia de senhas com BCrypt

#### Spring Data JPA

- Abstração para acesso a dados
- Hibernate como implementação JPA
- Queries automáticas e customizadas
- Transações declarativas

#### Java 17

- Linguagem de programação principal
- LTS (Long Term Support)
- Performance e recursos modernos
- Compatibilidade empresarial

#### Lombok

- Redução de código boilerplate
- Geração automática de getters/setters
- Construtores e builders automáticos
- Melhoria na legibilidade do código

### 4.2.2 Frontend

#### Thymeleaf

- Template engine server-side
- Integração nativa com Spring
- Sintaxe natural HTML
- Processamento no servidor

#### HTML5 + CSS3

- Estrutura semântica moderna
- Estilos responsivos
- Compatibilidade cross-browser
- Performance otimizada

#### JavaScript

- Interatividade client-side
- Validações dinâmicas
- AJAX para requisições assíncronas
- Manipulação do DOM

### 4.2.3 Banco de Dados

#### MySQL 8.0

- Sistema de gerenciamento relacional
- Performance e confiabilidade
- Suporte a transações ACID
- Escalabilidade horizontal e vertical

#### Características da Implementação

- Encoding UTF-8 para suporte internacional
- Índices otimizados para consultas frequentes
- Constraints para integridade referencial
- Triggers para auditoria automática

### 4.2.4 Ferramentas de Desenvolvimento

#### Maven

- Gerenciamento de dependências
- Build automatizado
- Ciclo de vida padronizado
- Integração com IDEs

#### Docker

- Containerização da aplicação
- Ambiente consistente
- Deploy simplificado
- Isolamento de dependências

### 4.2.5 Bibliotecas Adicionais

#### iText/OpenPDF

- Geração de relatórios em PDF
- Documentos com formatação profissional

- Integração com dados do sistema

## Spring Mail

- Envio de emails automático
- Suporte a templates HTML
- Configuração SMTP flexível

**ZXing**

- Geração de códigos QR
- Leitura de códigos de barras
- Integração com produtos

## Hibernate Validator

- Validação de dados
- Anotações declarativas
- Mensagens customizáveis

### 4.3 Estrutura de Pastas

### 4.3.1 Estrutura do Projeto

```

erp-corporativo/
├── .mvn/                                     # Configurações Maven
│   └── wrapper/
│       └── maven-wrapper.properties
├── src/
│   ├── main/
│   │   ├── java/
│   │   │   └── com/jaasielsilva/portaliceo/
│   │   │       ├── PortalCeoApplication.java    # Classe principal
│   │   │       ├── config/                      # Configurações
│   │   │           ├── DateConfig.java
│   │   │           ├── GlobalControllerAdvice.java
│   │   │           ├── MultipartConfig.java
│   │   │           └── SecurityConfig.java
│   │   │       ├── controller/                  # Controllers MVC
│   │   │           ├── DashboardController.java
│   │   │           ├── UsuarioController.java
│   │   │           ├── ClienteController.java
│   │   │           ├── ProdutoController.java
│   │   │           ├── EstoqueController.java
│   │   │           ├── VendaController.java
│   │   │           ├── FornecedorController.java
│   │   │           ├── RhController.java
│   │   │           ├── ContratosController.java
│   │   │           ├── SolicitacoesController.java
│   │   │           ├── PerfilController.java
│   │   │           ├── PermissaoController.java
│   │   │           ├── LoginController.java
│   │   │           ├── SenhaController.java
│   │   │           └── CustomErrorController.java
│   │   │       ├── dto/                          # Data Transfer Objects

```

```

├─ EstatisticasUsuariosDTO.java
├─ FornecedorDTO.java
├─ MovimentacaoEstoqueDTO.java
├─ UsuarioFormDTO.java
├─ UsuarioPDFDTO.java
├─ exception/                                # Tratamento de exceções
├─   ├─ BusinessException.java
├─   ├─ CargoNotFoundException.java
├─   ├─ ColaboradorNotFoundException.java
├─   ├─ DepartamentoNotFoundException.java
├─   └─ GlobalExceptionHandler.java
├─ model/                                    # Entidades JPA
├─   ├─ Usuario.java
├─   ├─ Cliente.java
├─   ├─ Produto.java
├─   ├─ Categoria.java
├─   ├─ Fornecedor.java
├─   ├─ Estoque.java
├─   ├─ Venda.java
├─   ├─ VendaItem.java
├─   ├─ Colaborador.java
├─   ├─ Cargo.java
├─   ├─ Departamento.java
├─   ├─ Contrato.java
├─   ├─ SolicitacaoAcesso.java
├─   ├─ Perfil.java
├─   ├─ Permissao.java
├─   └─ NivelAcesso.java
├─ repository/                              # Repositórios JPA
├─   ├─ UsuarioRepository.java
├─   ├─ ClienteRepository.java
├─   ├─ ProdutoRepository.java
├─   ├─ CategoriaRepository.java
├─   ├─ FornecedorRepository.java
├─   ├─ VendaRepository.java
├─   ├─ ColaboradorRepository.java
├─   ├─ CargoRepository.java
├─   ├─ DepartamentoRepository.java
├─   ├─ ContratoRepository.java
├─   ├─ SolicitacaoAcessoRepository.java
├─   ├─ PerfilRepository.java
├─   └─ PermissaoRepository.java
├─ security/                                # Configurações de segurança
├─   ├─ CustomAuthenticationFailureHandler.java
├─   └─ UsuarioDetailsService.java
└─ service/                                  # Serviços de negócio
    ├─ UsuarioService.java
    ├─ ClienteService.java
    ├─ ProdutoService.java
    ├─ CategoriaService.java
    ├─ FornecedorService.java
    └─ EstoqueService.java

```

```

├── VendaService.java
├── ColaboradorService.java
├── CargoService.java
├── DepartamentoService.java
├── ContratoService.java
├── SolicitacaoAcessoService.java
├── PerfilService.java
├── PermissaoService.java
├── EmailService.java
├── resources/
│   ├── application.properties          # Configurações da aplicação
│   ├── static/                        # Recursos estáticos
│   │   ├── css/                      # Folhas de estilo
│   │   │   ├── style.css
│   │   │   ├── login.css
│   │   │   ├── cliente-cadastro.css
│   │   │   ├── produto-form.css
│   │   │   └── ...
│   │   ├── js/                      # Scripts JavaScript
│   │   │   ├── script.js
│   │   │   ├── sidebar.js
│   │   │   ├── cliente-cadastro.js
│   │   │   └── ...
│   │   └── img/                     # Imagens
│   │       ├── logo.png
│   │       └── ...
│   └── templates/                   # Templates Thymeleaf
│       ├── components/              # Componentes reutilizáveis
│       │   ├── sidebar.html
│       │   ├── topbar.html
│       │   └── footer.html
│       ├── dashboard/
│       │   └── index.html
│       ├── usuarios/
│       │   ├── listar.html
│       │   ├── cadastro.html
│       │   ├── editar.html
│       │   └── detalhes.html
│       ├── clientes/
│       │   ├── lista.html
│       │   ├── cadastro.html
│       │   ├── editar.html
│       │   └── detalhes.html
│       ├── produtos/
│       ├── estoque/
│       ├── vendas/
│       ├── fornecedor/
│       ├── rh/
│       ├── contratos/
│       ├── solicitacoes/
│       ├── error/
│       │   └── 400.html

```

```
| | | | 401.html  
| | | | 403.html  
| | | | 404.html  
| | | | 500.html  
| | | └─ login.html  
| └─ test/ # Testes  
|   └─ java/  
|     └─ com/jaasielsilva/portaliceo/  
├─ target/ # Arquivos compilados  
├─ docker-compose.yml # Configuração Docker  
├─ dockerfile # Imagem Docker  
├─ pom.xml # Configuração Maven  
├─ README.md # Documentação básica  
├─ mvnw # Maven Wrapper (Unix)  
├─ mvnw.cmd # Maven Wrapper (Windows)  
└─ .gitignore # Arquivos ignorados pelo Git
```

### 4.3.2 Convenções de Nomenclatura

## Pacotes Java

- `controller` : Controllers MVC e REST
- `service` : Lógica de negócio
- `repository` : Acesso a dados
- `model` : Entidades JPA
- `dto` : Objetos de transferência
- `config` : Configurações
- `security` : Segurança
- `exception` : Tratamento de exceções

## Classes

- Controllers: `*Controller.java`
- Services: `*Service.java`
- Repositories: `*Repository.java`
- Models: Nome da entidade (ex: `Usuario.java` )
- DTOs: `*DTO.java`
- Exceptions: `*Exception.java`

## Templates

- Estrutura: `modulo/acao.html`
- Componentes: `components/nome.html`
- Páginas de erro: `error/codigo.html`

## Recursos Estáticos

- CSS: nome-modulo.css
- JavaScript: nome-modulo.js
- Imagens: formato descritivo

## 4.4 Endpoints REST

#### 4.4.1 Autenticação e Autorização

## Login

```
POST /login
Content-Type: application/x-www-form-urlencoded

username=usuario@email.com&password=senha123
```

### Resposta de Sucesso:

```
HTTP/1.1 302 Found
Location: /dashboard
Set-Cookie: JSESSIONID=...
```

### Resposta de Erro:

```
HTTP/1.1 302 Found
Location: /login?error=true
```

## Logout

```
POST /logout
```

### Resposta:

```
HTTP/1.1 302 Found
Location: /login?logout=true
```

## 4.4.2 Usuários

### Listar Usuários

```
GET /usuarios
Authorization: Required (COORDENADOR+)
```

### Parâmetros de Query:

- `busca` (opcional): Filtro por nome ou email

### Resposta:

```
<!-- Página HTML com lista de usuários -->
```

### Cadastrar Usuário

```
POST /usuarios/cadastrar
Content-Type: application/x-www-form-urlencoded
Authorization: Required (ADMIN+)
```

```
nome=João Silva&email=joao@empresa.com&senha=123456&confirmSenha=123456&perfilId=2
```

#### Resposta de Sucesso:

```
HTTP/1.1 302 Found
Location: /dashboard
```

#### Editar Usuário

```
POST /usuarios/{id}/editar
Content-Type: application/x-www-form-urlencoded
Authorization: Required (ADMIN+)

nome=João Silva Santos&email=joao.santos@empresa.com
```

#### Detalhes do Usuário

```
GET /usuarios/{id}/detalhes
Authorization: Required (COORDENADOR+)
```

#### Relatório de Usuários (PDF)

```
GET /usuarios/relatorio-pdf
Authorization: Required (GERENTE+)
```

#### Resposta:

```
HTTP/1.1 200 OK
Content-Type: application/pdf
Content-Disposition: attachment; filename="relatorio-usuarios.pdf"

[Conteúdo PDF]
```

### 4.4.3 Clientes

#### Listar Clientes

```
GET /clientes
Authorization: Required (OPERACIONAL+)
```

#### Parâmetros de Query:

- busca (opcional): Filtro por nome ou email

#### Cadastrar Cliente



POST /clientes/salvar

**Content-Type:** application/x-www-form-urlencoded

**Authorization:** Required (OPERACIONAL+)

nome=Empresa ABC&email=contato@empresaabc.com&tipoCliente=PJ&cpfCnpj=12.345.678/0001-90

### Editar Cliente

POST /clientes/{id}/editar

**Content-Type:** application/x-www-form-urlencoded

**Authorization:** Required (ANALISTA+)

### Detalhes do Cliente

GET /clientes/{id}/detalhes

**Authorization:** Required (OPERACIONAL+)

### Excluir Cliente (Lógico)

POST /clientes/{id}/excluir

**Content-Type:** application/json

**Authorization:** Required (ADMIN+)

**X-Matricula:** MATRICULA\_DO\_USUARIO

### Resposta de Sucesso:

```
{
  "mensagem": "Cliente excluído logicamente com sucesso."
}
```

### Resposta de Erro:

```
{
  "erro": "Matrícula incorreta para este usuário."
}
```

### Filtrar Clientes por Status

GET /clientes/filtro?status=ativo

**Authorization:** Required (OPERACIONAL+)

### Resposta:

```
[
  {
    "id": 1,
```

```
"nome": "Cliente Ativo",
"email": "cliente@email.com",
"status": "Ativo"
}
]
```

#### 4.4.4 Produtos

##### Listar Produtos

```
GET /produto/lista
Authorization: Required (OPERACIONAL+)
```

##### Cadastrar Produto

```
POST /produto/salvar
Content-Type: application/x-www-form-urlencoded
Authorization: Required (ANALISTA+)

nome=Produto Teste&preco=99.90&estoque=100&minimoEstoque=10
```

##### Buscar Produtos (API)

```
GET /api/produtos/buscar?termo=notebook
Authorization: Required (OPERACIONAL+)
```

##### Resposta:

```
[
  {
    "id": 1,
    "nome": "Notebook Dell",
    "preco": 2500.00,
    "estoque": 5
  }
]
```

#### 4.4.5 Estoque

##### Registrar Entrada

```
POST /estoque/entrada
Content-Type: application/x-www-form-urlencoded
Authorization: Required (ANALISTA+)

produtoId=1&quantidade=50&motivo=Compra&observacoes=Lote 123
```

### Registrar Saída

```
POST /estoque/saida
Content-Type: application/x-www-form-urlencoded
Authorization: Required (ANALISTA+)

produtoId=1&quantidade=10&motivo=Venda&observacoes=Pedido 456
```

### Listar Movimentações

```
GET /estoque/movimentacoes
Authorization: Required (ANALISTA+)
```

### Alertas de Estoque

```
GET /estoque/alertas
Authorization: Required (SUPERVISOR+)
```

## 4.4.6 Vendas

### Cadastrar Venda

```
POST /vendas/salvar
Content-Type: application/x-www-form-urlencoded
Authorization: Required (OPERACIONAL+)

clienteId=1&produtos[0].id=1&produtos[0].quantidade=2&produtos[0].preco=99.90
```

### Listar Vendas

```
GET /vendas/lista
Authorization: Required (OPERACIONAL+)
```

## 4.4.7 Solicitações

### Nova Solicitação

```
POST /solicitacoes/salvar
Content-Type: application/x-www-form-urlencoded
Authorization: Required (Todos)

nome=João Silva&email=joao@empresa.com&justificativa=Preciso acessar o
sistema&nivelAcesso=OPERACIONAL
```

### Aprovar Solicitação

```
POST /solicitacoes/{id}/aprovar
Content-Type: application/x-www-form-urlencoded
Authorization: Required (SUPERVISOR+)

comentarios=Aprovado conforme política da empresa
```

### Rejeitar Solicitação

```
POST /solicitacoes/{id}/rejeitar
Content-Type: application/x-www-form-urlencoded
Authorization: Required (SUPERVISOR+)

comentarios=Não atende aos critérios necessários
```

## 4.4.8 Dashboard

### Dados do Dashboard

```
GET /dashboard
Authorization: Required (Todos)
```

#### Resposta:

```
<!-- Página HTML com dados do dashboard -->
```

### API de Estatísticas

```
GET /api/dashboard/estatisticas
Authorization: Required (ANALISTA+)
```

#### Resposta:

```
{
  "totalClientes": 150,
  "totalVendas": 25000.00,
  "produtosEstoque": 500,
  "vendasMes": [
    {"mes": "Janeiro", "valor": 5000.00},
    {"mes": "Fevereiro", "valor": 6000.00}
  ]
}
```

## 4.4.9 Relatórios

### Relatório de Vendas (PDF)

GET /relatorios/vendas/pdf?inicio=2024-01-01&fim=2024-01-31  
**Authorization:** Required (ANALISTA+)

Relatório de Estoque (Excel)

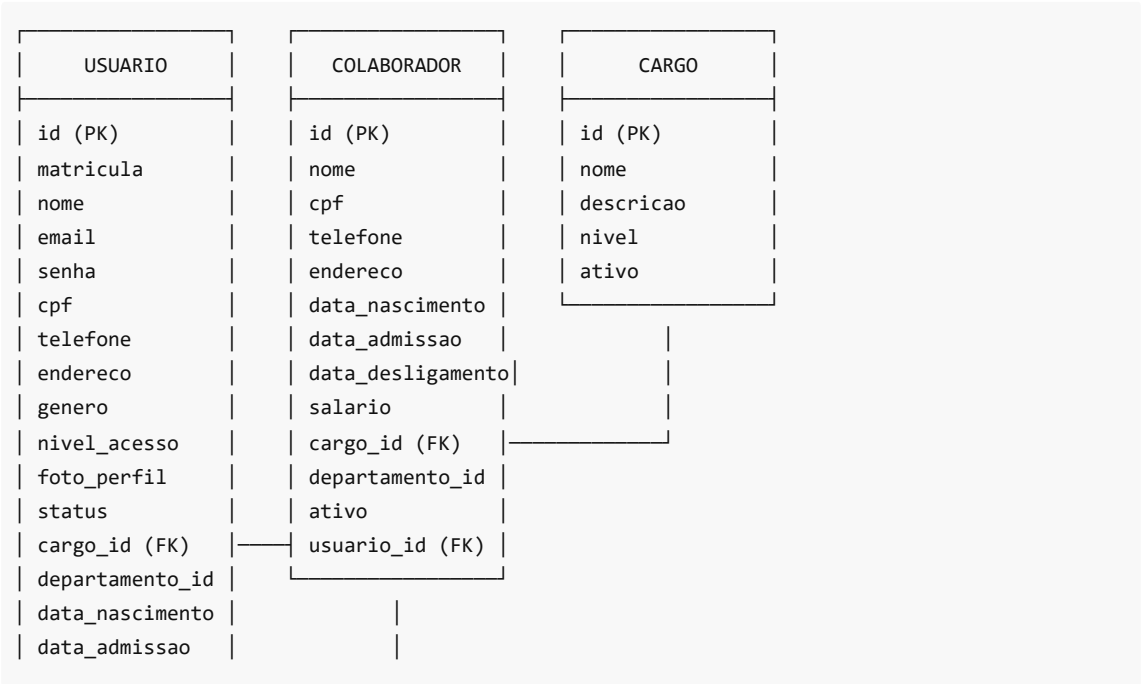
GET /relatorios/estoque/excel  
**Authorization:** Required (SUPERVISOR+)

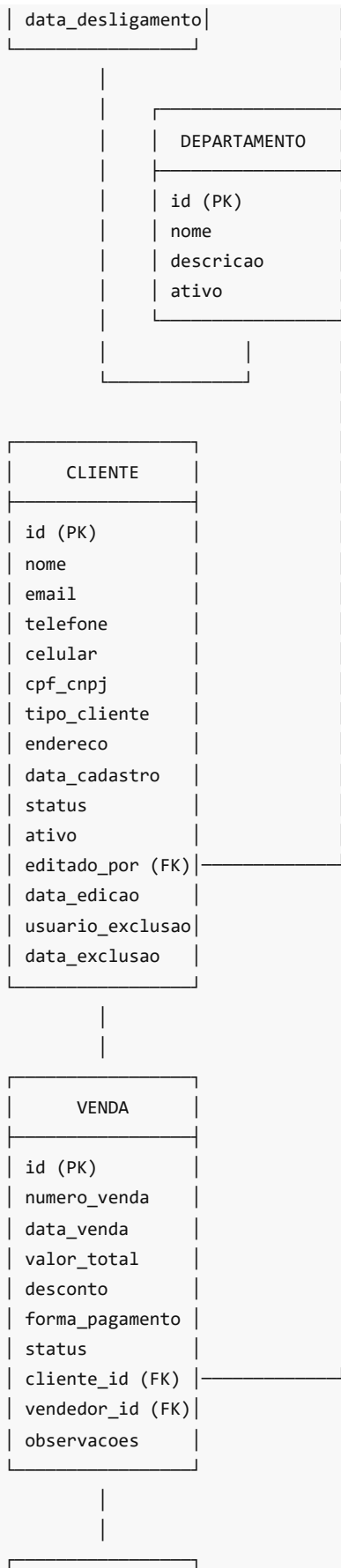
4.4.10 Códigos de Status HTTP

Código	Descrição	Uso
200	OK	Requisição bem-sucedida
201	Created	Recurso criado com sucesso
302	Found	Redirecionamento
400	Bad Request	Dados inválidos
401	Unauthorized	Não autenticado
403	Forbidden	Sem permissão
404	Not Found	Recurso não encontrado
500	Internal Server Error	Erro interno do servidor

4.5 Modelo de Banco de Dados

4.5.1 Diagrama Entidade-Relacionamento





VENDA_ITEM
id (PK)
quantidade
preco_unitario
subtotal
venda_id (FK)
produto_id (FK)

PRODUTO
id (PK)
nome
ean
codigo_interno
descricao
preco
estoque
minimo_estoque
unidade_medida
marca
peso
largura
altura
profundidade
imagem_url
ativo
data_cadastro
categoria_id (FK)
fornecedor_id

CATEGORIA
id (PK)
nome
descricao
ativo

FORNECEDOR
id (PK)
razao_social
nome_fantasia
cnpj

email	
telefone	
endereco	
contato	
ativo	
data_cadastro	

## 4.5.2 Principais Entidades

### Usuario

- **Propósito:** Controle de acesso ao sistema
- **Relacionamentos:**
  - ManyToOne com Cargo
  - ManyToOne com Departamento
  - OneToOne com Colaborador
  - ManyToMany com Perfil

### Cliente

- **Propósito:** Gestão de clientes da empresa
- **Características:** Suporte a PF e PJ, exclusão lógica
- **Relacionamentos:** OneToMany com Venda

### Produto

- **Propósito:** Catálogo de produtos
- **Características:** Controle de estoque, dimensões físicas
- **Relacionamentos:**
  - ManyToOne com Categoria
  - ManyToOne com Fornecedor

### Venda

- **Propósito:** Registro de vendas
- **Características:** Múltiplos itens, diferentes formas de pagamento
- **Relacionamentos:**
  - ManyToOne com Cliente
  - OneToMany com Vendaltem

## 4.5.3 Índices e Performance

### Índices Principais

```
-- Usuários
CREATE INDEX idx_usuario_email ON usuario(email);
CREATE INDEX idx_usuario_matricula ON usuario(matricula);
CREATE INDEX idx_usuario_status ON usuario(status);

-- Clientes
CREATE INDEX idx_cliente_cpf_cnpj ON cliente(cpf_cnpj);
CREATE INDEX idx_cliente_email ON cliente(email);
CREATE INDEX idx_cliente_ativo ON cliente(ativo);
```



```
-- Produtos
CREATE INDEX idx_produto_codigo ON produto(codigo_interno);
CREATE INDEX idx_produto_ean ON produto(ean);
CREATE INDEX idx_produto_categoria ON produto(categoria_id);

-- Vendas
CREATE INDEX idx_venda_data ON venda(data_venda);
CREATE INDEX idx_venda_cliente ON venda(cliente_id);
CREATE INDEX idx_venda_status ON venda(status);
```

#### 4.5.4 Constraints e Validações

##### Constraints de Integridade

```
-- Email único para usuários
ALTER TABLE usuario ADD CONSTRAINT uk_usuario_email UNIQUE (email);

-- CPF/CNPJ único para clientes
ALTER TABLE cliente ADD CONSTRAINT uk_cliente_cpf_cnpj UNIQUE (cpf_cnpj);

-- Código interno único para produtos
ALTER TABLE produto ADD CONSTRAINT uk_produto_codigo UNIQUE (codigo_interno);

-- Validações de domínio
ALTER TABLE cliente ADD CONSTRAINT ck_cliente_tipo
CHECK (tipo_cliente IN ('PF', 'PJ'));

ALTER TABLE usuario ADD CONSTRAINT ck_usuario_nivel
CHECK (nivel_acesso IN ('MASTER', 'ADMIN', 'GERENTE', 'COORDENADOR',
'SUPERVISOR', 'ANALISTA', 'OPERACIONAL', 'ESTAGIARIO'));
```

## 4.6 Instalação e Configuração

### 4.6.1 Pré-requisitos

#### Software Necessário

- **Java 17** ou superior (OpenJDK ou Oracle JDK)
- **MySQL 8.0** ou superior
- **Maven 3.6** ou superior
- **Git** para controle de versão

#### Verificação dos Pré-requisitos

```
# Verificar Java
java -version

# Verificar Maven
mvn -version

# Verificar MySQL
```

```
mysql --version
```

```
# Verificar Git
```

```
git --version
```

## 4.6.2 Configuração do Banco de Dados

### 1. Criar Banco de Dados

```
-- Conectar ao MySQL como root
```

```
mysql -u root -p
```

```
-- Criar banco de dados
```

```
CREATE DATABASE painelceo CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
```

```
-- Criar usuário (opcional)
```

```
CREATE USER 'erp_user'@'localhost' IDENTIFIED BY 'senha_segura';
```

```
GRANT ALL PRIVILEGES ON painelceo.* TO 'erp_user'@'localhost';
```

```
FLUSH PRIVILEGES;
```

### 2. Configurar Conexão

Editar `src/main/resources/application.properties` :

```
# Configuração do banco de dados
```

```
spring.datasource.url=jdbc:mysql://localhost:3306/painelceo
```

```
spring.datasource.username=root
```

```
spring.datasource.password=12345
```

```
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
```

```
# Configuração do Hibernate
```

```
spring.jpa.hibernate.ddl-auto=update
```

```
spring.jpa.database-platform=org.hibernate.dialect.MySQL8Dialect
```

```
spring.jpa.show-sql=false
```

```
spring.jpa.properties.hibernate.format_sql=true
```

## 4.6.3 Instalação Local

### 1. Clonar o Repositório

```
git clone https://github.com/usuario/erp-corporativo.git
```

```
cd erp-corporativo
```

### 2. Configurar Variáveis de Ambiente

```
# Windows
```

```
set JAVA_HOME=C:\Program Files\Java\jdk-17
```

```
set MAVEN_HOME=C:\Program Files\Apache\maven
```

```
set PATH=%JAVA_HOME%\bin;%MAVEN_HOME%\bin;%PATH%
```

```
# Linux/Mac
export JAVA_HOME=/usr/lib/jvm/java-17-openjdk
export MAVEN_HOME=/opt/maven
export PATH=$JAVA_HOME/bin:$MAVEN_HOME/bin:$PATH
```

### 3. Instalar Dependências

```
mvn clean install
```

### 4. Executar a Aplicação

```
# Modo desenvolvimento
mvn spring-boot:run

# Ou compilar e executar JAR
mvn clean package
java -jar target/portal-ceo-1.0.0.jar
```

### 5. Acessar a Aplicação

- URL: `http://localhost:8080`
- Login padrão: `admin@teste.com`
- Senha padrão: `123456`

#### 4.6.4 Configuração de Email

##### Gmail SMTP

```
# Configuração de email
spring.mail.host=smtp.gmail.com
spring.mail.port=587
spring.mail.username=seu-email@gmail.com
spring.mail.password=sua-senha-app
spring.mail.properties.mail.smtp.auth=true
spring.mail.properties.mail.smtp.starttls.enable=true
spring.mail.properties.mail.smtp.timeout=5000
```

##### Outros Provedores

```
# Outlook/Hotmail
spring.mail.host=smtp-mail.outlook.com
spring.mail.port=587

# Yahoo
spring.mail.host=smtp.mail.yahoo.com
spring.mail.port=587

# SMTP Corporativo
```

```
spring.mail.host=smtp.empresa.com
spring.mail.port=25
```

## 4.6.5 Configurações de Segurança

### Senhas Seguras

```
# Configuração de criptografia
app.security.password.strength=10
app.security.jwt.secret=chave-secreta-muito-longa-e-segura
app.security.session.timeout=3600
```

### HTTPS (Produção)

```
# Certificado SSL
server.ssl.key-store=classpath:keystore.p12
server.ssl.key-store-password=senha-keystore
server.ssl.key-store-type=PKCS12
server.ssl.key-alias=tomcat
server.port=8443
```

## 4.6.6 Configurações de Performance

### Pool de Conexões

```
# HikariCP (padrão do Spring Boot)
spring.datasource.hikari.maximum-pool-size=20
spring.datasource.hikari.minimum-idle=5
spring.datasource.hikari.idle-timeout=300000
spring.datasource.hikari.max-lifetime=600000
spring.datasource.hikari.connection-timeout=30000
```

### Cache

```
# Cache de segundo nível
spring.jpa.properties.hibernate.cache.use_second_level_cache=true
spring.jpa.properties.hibernate.cache.region.factory_class=org.hibernate.cache.ehcache.EhCacheF
```

## 4.7 Deploy em Produção

### 4.7.1 Preparação para Produção

#### 1. Profile de Produção

Criar `application-prod.properties` :

```
# Configurações de produção
spring.profiles.active=prod
```

```
# Banco de dados de produção
spring.datasource.url=jdbc:mysql://db-server:3306/painelceo_prod
spring.datasource.username=${DB_USERNAME}
spring.datasource.password=${DB_PASSWORD}

# Configurações de segurança
spring.jpa.hibernate.ddl-auto=validate
spring.jpa.show-sql=false

# Logs
logging.level.root=WARN
logging.level.com.jaasielsilva.portalceo=INFO
logging.file.name=/var/log/portal-ceo/application.log

# Server
server.port=8080
server.address=0.0.0.0
```

## 2. Build de Produção

```
# Compilar para produção
mvn clean package -Pprod

# Ou com profile específico
mvn clean package -Dspring.profiles.active=prod
```

### 4.7.2 Deploy com Docker

#### Dockerfile

```
FROM openjdk:17-jdk-slim

# Criar usuário não-root
RUN addgroup --system spring && adduser --system spring --ingroup spring

# Diretório de trabalho
WORKDIR /app

# Copiar JAR
COPY target/portal-ceo-*.jar app.jar

# Permissões
RUN chown spring:spring app.jar
USER spring:spring

# Porta
EXPOSE 8080
```

```
# Comando de execução
ENTRYPOINT ["java", "-jar", "app.jar"]
```

## **docker-compose.yml**

```
version: '3.8'

services:
  app:
    build: .
    ports:
      - "8080:8080"
    environment:
      - SPRING_PROFILES_ACTIVE=prod
      - DB_HOST=mysql
      - DB_USERNAME=erp_user
      - DB_PASSWORD=senha_segura
    depends_on:
      - mysql
    restart: unless-stopped
    volumes:
      - ./logs:/var/log/portal-ceo

  mysql:
    image: mysql:8.0
    environment:
      - MYSQL_ROOT_PASSWORD=root_password
      - MYSQL_DATABASE=painelceo_prod
      - MYSQL_USER=erp_user
      - MYSQL_PASSWORD=senha_segura
    ports:
      - "3306:3306"
    volumes:
      - mysql_data:/var/lib/mysql
      - ./init.sql:/docker-entrypoint-initdb.d/init.sql
    restart: unless-stopped

  nginx:
    image: nginx:alpine
    ports:
      - "80:80"
      - "443:443"
    volumes:
      - ./nginx.conf:/etc/nginx/nginx.conf
      - ./ssl:/etc/nginx/ssl
    depends_on:
      - app
    restart: unless-stopped
```

```
volumes:
  mysql_data:
```

## Comandos Docker

```
# Build e execução
docker-compose up -d

# Verificar logs
docker-compose logs -f app

# Parar serviços
docker-compose down

# Atualizar aplicação
docker-compose pull
docker-compose up -d --force-recreate
```

### 4.7.3 Deploy em Servidor Linux

#### 1. Preparar Servidor

```
# Atualizar sistema
sudo apt update && sudo apt upgrade -y

# Instalar Java 17
sudo apt install openjdk-17-jdk -y

# Instalar MySQL
sudo apt install mysql-server -y

# Configurar firewall
sudo ufw allow 8080
sudo ufw allow 22
sudo ufw enable
```

#### 2. Criar Usuário de Sistema

```
# Criar usuário para a aplicação
sudo useradd -r -s /bin/false portalceo
sudo mkdir /opt/portalceo
sudo chown portalceo:portalceo /opt/portalceo
```

#### 3. Configurar Systemd Service

Criar `/etc/systemd/system/portalceo.service` :

```
[Unit]
Description=Portal CEO ERP
```

```
After=network.target mysql.service

[Service]
Type=simple
User=portalceo
Group=portalceo
ExecStart=/usr/bin/java -jar /opt/portalceo/portal-ceo.jar
WorkingDirectory=/opt/portalceo
Restart=always
RestartSec=10

# Variáveis de ambiente
Environment=SPRING_PROFILES_ACTIVE=prod
Environment=JAVA_OPTS="-Xmx1024m -Xms512m"

# Logs
StandardOutput=journal
StandardError=journal

[Install]
WantedBy=multi-user.target
```

#### 4. Iniciar Serviço

```
# Recarregar systemd
sudo systemctl daemon-reload

# Habilitar inicialização automática
sudo systemctl enable portalceo

# Iniciar serviço
sudo systemctl start portalceo

# Verificar status
sudo systemctl status portalceo

# Ver logs
sudo journalctl -u portalceo -f
```

#### 4.7.4 Configuração de Proxy Reverso (Nginx)

##### nginx.conf

```
server {
    listen 80;
    server_name seu-dominio.com;
    return 301 https://$server_name$request_uri;
}

server {
```



```

listen 443 ssl http2;
server_name seu-dominio.com;

# SSL
ssl_certificate /etc/nginx/ssl/cert.pem;
ssl_certificate_key /etc/nginx/ssl/key.pem;
ssl_protocols TLSv1.2 TLSv1.3;
ssl_ciphers HIGH:!aNULL:!MD5;

# Proxy para aplicação
location / {
    proxy_pass http://localhost:8080;
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;

    # Timeouts
    proxy_connect_timeout 30s;
    proxy_send_timeout 30s;
    proxy_read_timeout 30s;
}

# Arquivos estáticos
location /static/ {
    alias /opt/portaiceo/static/;
    expires 1y;
    add_header Cache-Control "public, immutable";
}

# Logs
access_log /var/log/nginx/portaiceo_access.log;
error_log /var/log/nginx/portaiceo_error.log;
}

```

## 4.7.5 Monitoramento e Backup

### Script de Backup

```

#!/bin/bash
# backup.sh

DATE=$(date +%Y%m%d_%H%M%S)
BACKUP_DIR="/backup/portaiceo"
DB_NAME="painelceo_prod"

# Criar diretório de backup
mkdir -p $BACKUP_DIR

# Backup do banco de dados
mysqldump -u root -p$MYSQL_ROOT_PASSWORD $DB_NAME > $BACKUP_DIR/db_$DATE.sql

```

```
# Backup dos arquivos de upload
tar -czf $BACKUP_DIR/uploads_$(date +%Y-%m-%d).tar.gz /opt/portalceo/uploads/

# Remover backups antigos (manter 30 dias)
find $BACKUP_DIR -name "*.sql" -mtime +30 -delete
find $BACKUP_DIR -name "*.tar.gz" -mtime +30 -delete

echo "Backup concluído: $(date +%Y-%m-%d)"
```

### Crontab para Backup Automático

```
# Editar crontab
crontab -e

# Backup diário às 2h da manhã
0 2 * * * /opt/scripts/backup.sh >> /var/log/backup.log 2>&1
```

### Monitoramento com Script

```
#!/bin/bash
# monitor.sh

# Verificar se aplicação está rodando
if ! systemctl is-active --quiet portalceo; then
    echo "ALERTA: Portal CEO não está rodando!"
    systemctl restart portalceo
fi

# Verificar uso de memória
MEM_USAGE=$(free | grep Mem | awk '{printf "%.0f", $3/$2 * 100.0}')
if [ $MEM_USAGE -gt 90 ]; then
    echo "ALERTA: Uso de memória alto: ${MEM_USAGE}%"
fi

# Verificar espaço em disco
DISK_USAGE=$(df / | tail -1 | awk '{print $5}' | sed 's/%//')
if [ $DISK_USAGE -gt 85 ]; then
    echo "ALERTA: Espaço em disco baixo: ${DISK_USAGE}%"
fi
```

## 5. ANEXOS

### 5.1 FAQ

#### 5.1.1 Perguntas Frequentes - Usuários

**P: Como alterar minha senha?** R: Acesse "Meu Perfil" no menu superior direito e clique em "Alterar Senha". Digite a senha atual e a nova senha duas vezes.

**P: Esqueci minha senha, como recuperar?** R: Na tela de login, clique em "Esqueci minha senha" e informe seu email. Você receberá instruções para criar uma nova senha.

**P: Por que não consigo acessar determinada funcionalidade?** R: Cada funcionalidade requer um nível de acesso específico. Verifique com seu gestor se você possui as permissões necessárias.

**P: Como cadastrar um novo cliente?** R: Acesse "Clientes" → "Cadastro", preencha todos os campos obrigatórios e clique em "Salvar". Certifique-se de que o CPF/CNPJ não esteja já cadastrado.

**P: O sistema está lento, o que fazer?** R: Verifique sua conexão com a internet. Se o problema persistir, contate o suporte técnico.

**P: Como gerar relatórios?** R: Acesse o módulo desejado e procure pela opção "Relatórios". Selecione o período e clique em "Gerar PDF" ou "Exportar Excel".

### 5.1.2 Perguntas Frequentes - Administradores

**P: Como criar um novo usuário?** R: Acesse "Usuários" → "Cadastro", preencha os dados e selecione o nível de acesso apropriado. O usuário receberá as credenciais por email.

**P: Como alterar o nível de acesso de um usuário?** R: Acesse "Usuários" → "Listar", encontre o usuário e clique em "Editar". Altere o nível de acesso e salve.

**P: Como fazer backup dos dados?** R: O backup é automático, mas você pode gerar backups manuais através do painel administrativo ou executando o script de backup.

**P: Como configurar alertas de estoque?** R: No cadastro de produtos, defina o "Estoque Mínimo". O sistema gerará alertas automaticamente quando o estoque atingir esse nível.

**P: Como integrar com outros sistemas?** R: O sistema possui APIs REST documentadas. Contate o suporte técnico para orientações sobre integração.

### 5.1.3 Perguntas Frequentes - Técnicas

**P: Quais são os requisitos mínimos do servidor?** R: 4GB RAM, 2 CPU cores, 50GB de armazenamento, Java 17, MySQL 8.0.

**P: Como atualizar o sistema?** R: Faça backup, pare o serviço, substitua o arquivo JAR e reinicie o serviço.

**P: Como configurar HTTPS?** R: Configure um certificado SSL no Nginx ou diretamente no Spring Boot através das propriedades SSL.

**P: Como monitorar a performance?** R: Use ferramentas como Prometheus + Grafana ou monitore logs do sistema e métricas do banco de dados.

## 5.2 Checklist de Segurança

### 5.2.1 Segurança da Aplicação

#### Autenticação e Autorização

- ☐ Senhas criptografadas com BCrypt
- ☐ Sessões com timeout configurado

- ☐ Controle de acesso baseado em roles
- ☐ Validação de permissões em todos os endpoints
- ☐ Proteção contra ataques de força bruta
- ☐ Logout seguro com invalidação de sessão

#### **Proteção contra Ataques**

- ☐ Proteção CSRF habilitada
- ☐ Headers de segurança configurados
- ☐ Validação de entrada em todos os formulários
- ☐ Sanitização de dados de saída
- ☐ Proteção contra SQL Injection (uso de JPA)
- ☐ Proteção contra XSS

#### **Configurações Seguras**

- ☐ Senhas padrão alteradas
- ☐ Usuários de teste removidos em produção
- ☐ Logs de segurança habilitados
- ☐ Informações sensíveis não expostas em logs
- ☐ Configurações de produção separadas

### **5.2.2 Segurança da Infraestrutura**

#### **Servidor**

- ☐ Sistema operacional atualizado
- ☐ Firewall configurado
- ☐ Serviços desnecessários desabilitados
- ☐ Usuários com privilégios mínimos
- ☐ Chaves SSH configuradas
- ☐ Acesso root restrito

#### **Banco de Dados**

- ☐ Usuário específico para aplicação
- ☐ Senhas fortes configuradas
- ☐ Acesso restrito por IP
- ☐ Backup criptografado
- ☐ Logs de auditoria habilitados

#### **Rede**

- ☐ HTTPS configurado
- ☐ Certificados SSL válidos
- ☐ VPN para acesso administrativo
- ☐ Monitoramento de tráfego
- ☐ IDS/IPS configurado

### **5.2.3 Políticas de Segurança**

#### **Senhas**

- Mínimo 8 caracteres
- Combinação de letras, números e símbolos
- Alteração obrigatória a cada 90 dias
- Não reutilização das últimas 5 senhas
- Bloqueio após 5 tentativas incorretas

#### **Acesso**

- Princípio do menor privilégio
- Revisão trimestral de permissões
- Desativação imediata de usuários desligados
- Auditoria mensal de acessos
- Segregação de funções

#### **Dados**

- Classificação de dados por sensibilidade
- Criptografia de dados sensíveis
- Backup diário com teste de restauração
- Retenção de logs por 1 ano
- Política de descarte seguro

## **5.3 Plano de Testes**

### **5.3.1 Testes Funcionais**

#### **Módulo de Autenticação**

##### **Teste 1: Login Válido**

- **Objetivo:** Verificar login com credenciais válidas
- **Passos:**
  1. Acessar página de login
  2. Inserir email e senha válidos
  3. Clicar em "Entrar"
- **Resultado Esperado:** Redirecionamento para dashboard

##### **Teste 2: Login Inválido**

- **Objetivo:** Verificar comportamento com credenciais inválidas
- **Passos:**
  1. Acessar página de login
  2. Inserir email ou senha inválidos
  3. Clicar em "Entrar"
- **Resultado Esperado:** Mensagem de erro exibida

##### **Teste 3: Logout**

- **Objetivo:** Verificar logout do sistema
- **Passos:**
  1. Fazer login no sistema
  2. Clicar em "Sair"
- **Resultado Esperado:** Redirecionamento para login

#### **Módulo de Usuários**

#### Teste 4: Cadastro de Usuário

- **Objetivo:** Verificar cadastro de novo usuário
- **Passos:**
  1. Acessar "Usuários" → "Cadastro"
  2. Preencher todos os campos obrigatórios
  3. Clicar em "Salvar"
- **Resultado Esperado:** Usuário criado com sucesso

#### Teste 5: Edição de Usuário

- **Objetivo:** Verificar edição de dados do usuário
- **Passos:**
  1. Acessar lista de usuários
  2. Clicar em "Editar" em um usuário
  3. Alterar dados e salvar
- **Resultado Esperado:** Dados atualizados corretamente

### Módulo de Clientes

#### Teste 6: Cadastro de Cliente PF

- **Objetivo:** Verificar cadastro de pessoa física
- **Passos:**
  1. Acessar "Clientes" → "Cadastro"
  2. Selecionar "Pessoa Física"
  3. Preencher dados e salvar
- **Resultado Esperado:** Cliente PF cadastrado

#### Teste 7: Cadastro de Cliente PJ

- **Objetivo:** Verificar cadastro de pessoa jurídica
- **Passos:**
  1. Acessar "Clientes" → "Cadastro"
  2. Selecionar "Pessoa Jurídica"
  3. Preencher dados e salvar
- **Resultado Esperado:** Cliente PJ cadastrado

### 5.3.2 Testes de Segurança

#### Teste 8: Controle de Acesso

- **Objetivo:** Verificar restrições de acesso por nível
- **Passos:**
  1. Login com usuário de nível baixo
  2. Tentar acessar funcionalidade restrita
- **Resultado Esperado:** Acesso negado (403)

#### Teste 9: Proteção CSRF

- **Objetivo:** Verificar proteção contra CSRF
- **Passos:**
  1. Tentar submeter formulário sem token CSRF
- **Resultado Esperado:** Requisição rejeitada

#### **Teste 10: Validação de Entrada**

- **Objetivo:** Verificar validação de dados
- **Passos:**
  1. Inserir dados inválidos em formulários
  2. Tentar salvar
- **Resultado Esperado:** Mensagens de validação exibidas

### **5.3.3 Testes de Performance**

#### **Teste 11: Carga de Usuários**

- **Objetivo:** Verificar comportamento com múltiplos usuários
- **Configuração:** 50 usuários simultâneos
- **Duração:** 10 minutos
- **Critério:** Tempo de resposta < 3 segundos

#### **Teste 12: Volume de Dados**

- **Objetivo:** Verificar performance com grande volume
- **Configuração:** 10.000 registros por tabela
- **Critério:** Consultas < 2 segundos

### **5.3.4 Testes de Integração**

#### **Teste 13: Integração com Banco**

- **Objetivo:** Verificar operações CRUD
- **Passos:**
  1. Criar registro
  2. Ler registro
  3. Atualizar registro
  4. Excluir registro
- **Resultado Esperado:** Todas as operações funcionando

#### **Teste 14: Envio de Email**

- **Objetivo:** Verificar envio de emails
- **Passos:**
  1. Cadastrar usuário
  2. Verificar recebimento de email
- **Resultado Esperado:** Email recebido corretamente

### **5.3.5 Testes de Usabilidade**

#### **Teste 15: Navegação**

- **Objetivo:** Verificar facilidade de navegação
- **Critério:** Usuário consegue completar tarefas básicas sem ajuda














#### **Teste 16: Responsividade**

- **Objetivo:** Verificar funcionamento em dispositivos móveis
- **Dispositivos:** Smartphone, tablet, desktop
- **Critério:** Interface adaptada corretamente

## **5.4 Registro de Mudanças**

## 5.4.1 Versão 1.0.0 (Data de Lançamento)

### Funcionalidades Implementadas

-  Sistema de autenticação e autorização
-  Módulo de gestão de usuários
-  Módulo de gestão de clientes
-  Módulo de gestão de produtos
-  Módulo de controle de estoque
-  Módulo de vendas
-  Módulo de fornecedores
-  Módulo de RH básico
-  Dashboard com indicadores
-  Sistema de relatórios em PDF
-  Controle de permissões por nível
-  Auditoria de ações
-  Interface responsiva

### Tecnologias Utilizadas







- Spring Boot 3.5.3
- Spring Security
- Spring Data JPA
- Thymeleaf
- MySQL 8.0
- Java 17
- Maven

### Configurações Iniciais






- Usuário administrador padrão
- Estrutura de banco de dados
- Configurações de email
- Perfis de desenvolvimento e produção

## 5.4.2 Roadmap - Próximas Versões



### Versão 1.1.0 (Planejada)

-  Módulo financeiro completo
-  Integração com APIs de pagamento
-  Relatórios avançados com gráficos
-  Notificações push
-  API REST completa
-  Módulo de compras




### Versão 1.2.0 (Planejada)

-  Integração com contabilidade
-  Módulo de projetos
-  Controle de ponto eletrônico
-  Chat interno
-  Aplicativo mobile

### Versão 2.0.0 (Futuro)

-  Microserviços
-  Inteligência artificial



-  Business Intelligence
-  Integração com e-commerce
-  Multi-tenancy

### 5.4.3 Correções e Melhorias

#### Bugs Conhecidos

- Nenhum bug crítico identificado

#### Melhorias Implementadas

- Otimização de consultas ao banco
- Melhoria na interface do usuário
- Validações mais robustas
- Logs mais detalhados
- Performance aprimorada

#### Problemas Resolvidos

- Correção na validação de CPF/CNPJ
- Ajuste no cálculo de comissões
- Melhoria no controle de sessões
- Correção em relatórios de estoque

---

## SUPORTE E CONTATO

### Suporte Técnico

- **Email:** [suporte@portalceo.com](mailto:suporte@portalceo.com)
- **Telefone:** (11) 9999-9999
- **Horário:** Segunda a Sexta, 8h às 18h

### Documentação

- **Site:** <https://docs.portalceo.com>
- **GitHub:** <https://github.com/usuario/erp-corporativo>
- **Wiki:** <https://wiki.portalceo.com>

### Treinamento

- **Agendamento:** [treinamento@portalceo.com](mailto:treinamento@portalceo.com)
- **Material:** <https://treinamento.portalceo.com>
- **Certificação:** Disponível após conclusão

---

© 2024 Portal do CEO - ERP Corporativo. Todos os direitos reservados.

Documento gerado automaticamente em: {{ data\_atual }} Versão da documentação: 1.0 Versão do sistema: 1.0.0