

# 1. Objetivo do Planejamento de Testes

Este documento define a estratégia de testes do sistema, estabelecendo **tipos de testes, priorização por risco, escopo de cobertura e organização dos casos de teste**, com base nos requisitos definidos no documento de requisitos.

---

## 2. Estratégia Geral de Testes

### 2.1 Abordagem

- Testes **funcionais manuais** para validação de regras de negócio e aderência ao Figma
  - Testes **automatizados (Cypress)** focados em fluxos críticos e regressão
  - Testes **exploratórios** para identificação de falhas de UX e inconsistências visuais
- 

## 3. Tipos de Teste Aplicáveis

### 3.1 Testes Funcionais

Validam se o sistema atende aos requisitos funcionais definidos.

Exemplos:

- Cadastro de funcionário com dados válidos
  - Edição e exclusão de funcionário
  - Cadastro de múltiplas atividades
- 

### 3.2 Testes de Validação de Dados

Focados na integridade das informações inseridas.

Exemplos:

- CPF válido com e sem pontuação
- CPF inválido
- RG com caracteres não permitidos

- Campos obrigatórios em branco
- 

### 3.3 Testes de Navegação e UX

Avaliam a experiência do usuário e o comportamento da interface.

Exemplos:

- Navegação por botões onde é espada a mensagem “Em Breve”
  - Feedback visual ao interagir com botões
  - Diferenciação visual entre funcionários ativos e inativos
  - Presença de barra de rolagem quando aplicável
- 

## 4. Priorização por Risco

### 4.1 Critérios de Risco

- **Alto:** Impede uso do sistema ou gera dados incorretos
- **Médio:** Impacta fluxo, mas possui alternativa
- **Baixo:** Impacto visual ou de usabilidade leve

### 4.2 Matriz de Prioridade

#### Risco Alto

- Cadastro de funcionário com falha
- Validação incorreta de CPF/RG
- Bug ao adicionar nova atividade

#### Risco Médio

- Botões sem ações
- Diferenciação visual incorreta de funcionários ativos e inativos

#### Risco Baixo

- Alinhamento visual
  - Estilização incompleta de botões
  - Imagens fora da posição exata
-

## 5. Ferramentas Utilizadas

- IDE: WebStorm
  - Automação: Cypress
  - Linguagem: JavaScript
  - Evidências: Prints de tela
- 

## 6. Considerações Finais

Este planejamento prioriza **qualidade funcional e aderência ao Figma**, focando nos pontos de maior risco identificados durante a análise exploratória do sistema.