## 1. Introdução

#### 1.1 Objetivo

Este documento define as estratégias e os casos de teste que serão utilizados para validar o sistema de gerenciamento de filas, agendamentos e documentos das USFs. O objetivo é garantir que o sistema atenda aos requisitos funcionais e não funcionais especificados.

#### 1.2 Escopo

O plano de testes abrange:

- Testes de unidade para funções individuais do backend.
- Testes de integração entre o frontend e o backend.
- Testes end-to-end para validação de fluxos completos no sistema.
- Testes de desempenho e segurança.

#### 1.3 Referências

- Documento de Requisitos.
- Documento de Especificação de Software.

## 2. Estratégia de Testes

#### 2.1 Ferramentas Utilizadas

- **Backend:**
    - Postman (testes de API).
    - Jest (testes unitários).
- **Frontend:**
    - Flutter Test (testes de widget e interface).
    - Selenium ou Cypress (testes end-to-end).
- **Integração:**
    - Newman (extensão do Postman para execução de coleções automatizadas).
- **Desempenho e Segurança:**
    - JMeter (testes de desempenho).
    - OWASP ZAP (verificação de vulnerabilidades).

#### 2.2 Tipos de Teste

1. **Teste de Unidade:**
    - Valida funções individuais, como geração de tokens e operações CRUD no MongoDB.
2. **Teste de Integração:**
    - Valida a comunicação entre o frontend e o backend.
3. **Teste End-to-End:**
    - Simula fluxos completos, como agendamento, validação de chegada e envio de justificativa.
4. **Teste de Desempenho:**
    - Avalia tempos de resposta e capacidade de lidar com carga elevada.
5. **Teste de Segurança:**
    - Verifica proteção de dados sensíveis e vulnerabilidades em APIs e banco de dados.

## 3. Casos de Teste

#### 3.1 Teste de Unidade

- **ID:** TU01
    - **Funcionalidade:** Geração de tokens únicos para validação de chegada.
    - **Entrada:** Solicitação de token com ID do paciente e ID da unidade.
    - **Saída Esperada:** Token alfanumérico exclusivo com validade de 15 minutos.
- **ID:** TU02
    - **Funcionalidade:** Operações CRUD na coleção `usuarios`.
    - **Entrada:** Dados de cadastro de um novo paciente.
    - **Saída Esperada:** Registro armazenado corretamente no MongoDB.

#### 3.2 Teste de Integração

- **ID:** TI01
    - **Funcionalidade:** Validação de chegada com token e geolocalização.
    - **Entrada:** Token válido e localização do dispositivo.

- **Saída Esperada:** Inclusão do paciente na fila correspondente.

#### 3.3 Teste End-to-End

- **ID:** TE01
    - **Funcionalidade:** Fluxo completo de agendamento e atendimento.
    - **Fluxo:**
        1. O paciente agenda uma consulta via app.
        2. Recebe notificações de lembrete.
        3. Valida a chegada com token.
        4. Realiza o atendimento e acessa os documentos gerados.
    - **Saída Esperada:** Fluxo executado sem erros.

#### 3.4 Teste de Desempenho

- **ID:** TD01
    - **Funcionalidade:** Tempo de resposta da API de agendamento.
    - **Métrica:** Resposta em menos de 3 segundos para 100 requisições simultâneas.

#### 3.5 Teste de Segurança

- **ID:** TS01
    - **Funcionalidade:** Proteção de dados sensíveis no MongoDB.
    - **Procedimento:** Simular ataque de injeção de SQL.
    - **Resultado Esperado:** Nenhum dado exposto ou alterado.

## 4. Cronograma de Testes

| Fase | Período | Atividade |
|---|---|---|
| Planejamento | Semana 1 | Definição de cenários e ferramentas |
| Testes de Unidade | Semana 2 | Implementação e execução |
| Testes de Integração | Semana 3 | Validação entre módulos |
| Testes End-to-End | Semana 4 | Simulação de fluxos completos |
| Testes de Desempenho e Segurança | Semana 5 | Avaliação de desempenho e proteção |

## 5. Critérios de Sucesso

- **Funcionalidade:** 100% dos casos de teste de unidade e integração aprovados.

- **Desempenho:** Tempo de resposta inferior a 3 segundos em todas as operações comuns.
- **Segurança:** Nenhuma vulnerabilidade crítica detectada.

### 6. Conclusão

Este plano de testes garante a validação abrangente do sistema, assegurando que ele atenda aos padrões de qualidade e segurança esperados. A aprovação será baseada na satisfação dos critérios definidos.