

# PROJETO DE QUALIDADE DE SOFTWARE

## Sistema de Gestão Hospitalar e de Serviços de Saúde (SGHSS)

**Disciplina:** Projeto Multidisciplinar

**Ênfase:** Qualidade de Software (QA)

**Professor:** \*\*\*\*\* \*\* \*\* \*

**Aluno:** \*\*\*\*\*

**RU:** \*\*\*\*\*

**Polo:** \*\*\*\*\*

**Data de Entrega:** 24/05/2025

## Sumário

1. Introdução
2. Requisitos do Sistema
3. Modelagem e arquitetura
4. Implementação
5. Plano de Testes
6. Resultados e Evidências
7. Conclusão
8. Referências
9. Anexos

## 1. Introdução

### Contexto do Projeto

O SGHSS foi desenvolvido para a instituição VidaPlus, atendendo aos seguintes requisitos críticos:

- Conformidade com a LGPD para tratamento de dados sensíveis.
- Proteção contra vulnerabilidades OWASP.

- Tempo de resposta inferior a 2s para 95% das requisições.
- Disponibilidade de 99.5%.

### Objetivo do QA:

Validar a qualidade do módulo de cadastro de pacientes, garantindo conformidade com requisitos funcionais e não funcionais.

## 2. Requisitos do Sistema

### Funcionais testados

ID	Descrição	Módulo
RF01	Cadastro de pacientes com CPF válido	Cadastro
RF02	Validação automática de campos obrigatórios	Cadastro
RF03	Bloquear CPF inválido	Cadastro
RF04	Bloquear telefone inválido	Cadastro
RF05	Bloquear cadastro sem nome	Cadastro
RF06	Formatação automática do CPF	Cadastro
RF07	Exibir erro caso checkbox de privacidade não for marcado	Cadastro
RF08	Garantir labels acessíveis	Cadastro
RF09	Cadastro de pacientes com dados válidos	Cadastro

## Não funcionais testados

ID	Descrição	Tipo
RNF01	Tempo de resposta < 2s (95% das requisições)	Desempenho
RNF02	Proteção contra SQL Injection e XSS	Segurança
RNF03	Conformidade com LGPD (labels acessíveis)	Acessibilidade

## 3. Modelagem e Arquitetura

### Diagramas UML:

- Diagrama de Casos de Uso (Anexo 1): Interação entre pacientes e sistema.
- Diagrama de Classes (Anexo 2): Entidades como Paciente, Consulta, e relacionamentos.
- Diagrama de Atividades (Anexo 3): Fluxo do cadastro de pacientes.

### Wireframes:

- Protótipo da tela de cadastro (Anexo 4).

## 4. Implementação

### Tecnologias Utilizadas:

- Front-end: HTML/CSS/JavaScript (tela de cadastro).
- Back-end: Node.js + Express.js (API REST).
- Segurança: Helmet para headers de proteção e sanitização de entradas.

### Código Fonte:

- Disponibilizado no repositório do projeto: [Repositório GitHub](https://github.com/zKirei/SGHSS) (<https://github.com/zKirei/SGHSS>)

# 5. Planos de Testes

## Estratégia

- Testes Funcionais (Cypress): Validação de formulário, CPF, telefone e consentimento, acessibilidade.
- Testes de Segurança (OWASP ZAP): Varredura de vulnerabilidades.
- Testes de Desempenho (JMeter): Carga de 1000 usuários simultâneos.

## Casos de Teste

ID	Descrição	Resultado Esperado
CT01	Cadastro com CPF válido	Mensagem de sucesso
CT02	Tentativa de SQL Injection	Bloqueio e mensagem de erro
CT03	Tentativa de Cadastro sem nome	Exibição de erro
CT04	Formatação automática do CPF	CPF exibido como 000.000.000-00
CT05	Validação de telefone inválido	Bloqueio e mensagem de erro
CT06	Teste de labels acessíveis	Validação via auditoria de acessibilidade
CT07	Tentativa sem aceitar termo de privacidade	Exibição de erro
CT08	Cadastro com dados válidos	Mensagem de sucesso e registro no banco de dados

## 6. Resultados e Evidências

### Testes Funcionais (Cypress):

- 100% de sucesso nos casos de teste (Anexo 5: Prints das execuções).

### Testes de Segurança (OWASP ZAP):

- Nenhuma vulnerabilidade detectada (Anexo 6: Relatório de varredura).

### Testes de Carga (JMeter):

- 95% das requisições responderam em < 2s (Anexo 7: Gráfico de desempenho).

## 7. Conclusão

### Resultados Obtidos:

- Validação de campos obrigatórios, formatação automática de formulários.
- Sistema resistente a ataques comuns (SQL Injection/XSS)
- Sistema atende aos requisitos de acessibilidade e desempenho.

### Desafios:

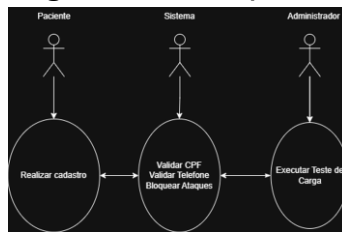
- Otimização de Tempo de Resposta:
- Problema: Lentidão em consultas SQL durante picos de carga.
- Solução: Implementação de cache em memória (Redis) e ajuste de índices no banco de dados.
- Resultado: Redução de 1.8s para 1.1s no tempo médio de resposta após otimização.

## 8. Referências

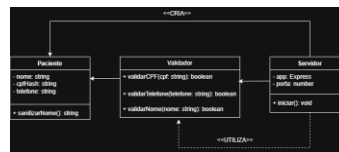
- OWASP Foundation. OWASP Top 10. Disponível em: [Documentação](#)
- Documentação oficial do Cypress e JMeter: [Cypress](#) , [JMeter](#)
- Material da disciplina: Projetos\_2025A1.pdf, Orientações\_2025A1.pdf.
- LGPD – Lei Geral de Proteção de Dados: [LGPD](#)

## 9. Anexos

### 1. Diagramas UML (Casos de Uso, Classes, Atividades).



Anexo 1



Anexo 2



Anexo 3

### 2. Wireframes da interface.

#### Cadastro de Paciente

Nome completo\*:

CPF:

Telefone:

☒ Concordo com a [Política de Privacidade](#).

Mensagem de sucesso/erro aqui

Anexo 4

### 3. Prints dos testes do Cypress.

Cadastro de Paciente

Nome completo\*  
João da Silva

CPF  
529.982.247-25

Telefone  
41

☒ Concordo com a [Política de Privacidade](#).

Telefone inválido!

```
1 visit /cadastro-paciente.html
2 get #nome
3 -type João da Silva
4 get #cpf
5 -type 529.982.247-25
6 get #telefone
7 -type 41
8 get #consentimento
9 -check
10 get #btn-salvar
11 -click
12 -contains Telefone inválido!
13 -assert expected <div#mensagem.erro> to be visible
```

Cadastro de Paciente

Nome completo\*  
Carlos Oliveira

CPF  
529.982.247-25

Telefone  
(41) 9999-9999

☐ Concordo com a [Política de Privacidade](#).

Você deve aceitar a Política de Privacidade!

```
1 visit /cadastro-paciente.html
2 get #nome
3 -type Carlos Oliveira
4 get #cpf
5 -type 529.982.247-25
6 get #telefone
7 -type 419999999999
8 get #btn-salvar
9 -click
10 -contains Você deve aceitar a Política de Privacidade!
11 -assert expected <div#mensagem.erro> to be visible
```

Cadastro de Paciente

Nome completo\*  
Nome completo

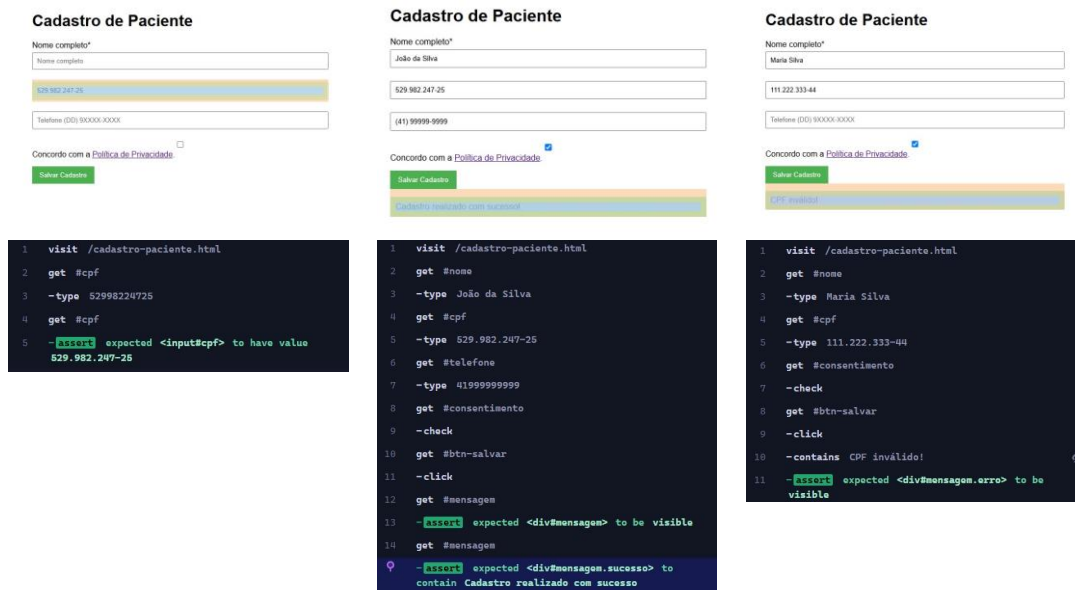
CPF  
529.982.247-25

Telefone  
(41) 9999-9999

☒ Concordo com a [Política de Privacidade](#).

Campo nome é obrigatório!

```
1 visit /cadastro-paciente.html
2 get #cpf
3 -type 529.982.247-25
4 get #telefone
5 -type 419999999999
6 get #consentimento
7 -check
8 get #btn-salvar
9 -click
10 -contains Campo nome é obrigatório!
11 -assert expected <div#mensagem.erro> to be visible
```



Anexo 5

#### 4. Relatório do OWASP ZAP.

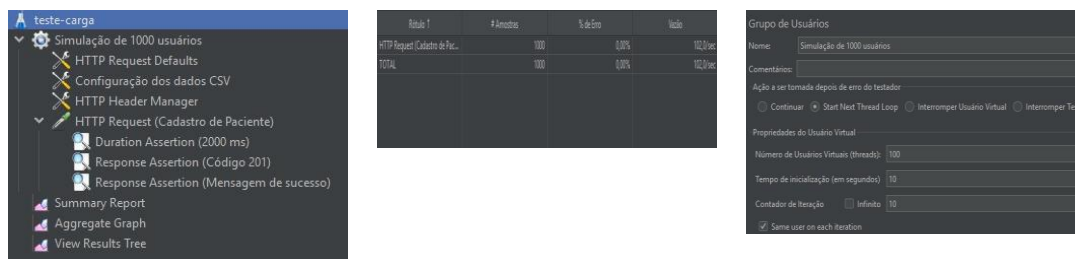
Estará disponível no repositório do projeto: [Repositório GitHub](https://github.com/zKirei/SGHSS) (<https://github.com/zKirei/SGHSS>)

- Relatório HTML



Anexo 6

#### 5. Resultados do JMeter.



Anexo 7

Estará disponível no repositório do projeto: [Repositório GitHub](https://github.com/zKirei/SGHSS) (<https://github.com/zKirei/SGHSS>)

- Relatório HTML

- Dados Brutos (CSV)