Documentação de Sprints – Projeto SaneaSP

# 1ª Sprint (17/03 – 01/04)

Objetivo:

* Preparar a interface do sistema para futura integração com backend.

Atividades realizadas:

* Atualizações nas telas já existentes.
* Estruturação inicial para a criação do backend e futura integração com o frontend.

Requisitos:

* RF002 (Gerenciamento e cadastro de reclamações)
* RF004 (Gerenciamento e cadastro de doenças)
* RNF003 (Responsividade)

# 2ª Sprint (01/04 – 16/04)

Objetivo:

* Expandir a interface do sistema com novas funcionalidades descobertas durante a definição do backend.

Atividades realizadas:

* Desenvolvimento de novas telas correspondentes às operações restantes do CRUD.
* Incluir acessibilidade nas interfaces.

Requisitos:

* RF002 (Gerenciamento e cadastro de reclamações)
* RF005 (Gerenciamento e cadastro de notícias)
* RNF003 (Responsividade)
* RNF006 (Acessibilidade)

# 3ª Sprint (17/04 – 28/04)

Objetivo:

* Iniciar o desenvolvimento das API’s do backend do projeto
* Iniciar estruturação do banco de dados utilizando Sequelize em SQlite

Atividades realizadas:

* Início da implementação da API do projeto (rotas e controllers).
* Desenvolvimento parcial do banco de dados com Sequelize.
* Criar as migrations do projeto

Requisitos:

* RF001 (Login e autenticação)
* RF002 (Gerenciamento e cadastro de reclamações)
* RNF002 (Banco de dados)
* RNF008 (Backend e API)

# 4ª Sprint (28/04 – 12/05)

Objetivo:

* Finalizar a estruturação do banco de dados com SQlite
* Iniciar integração da API com os models

Atividades realizadas:

* Finalização do banco de dados utilizando Sequelize.
* Integração do banco com a API desenvolvida anteriormente.
* Início da integração com o FrontEnd
* Criar as models do ORM para manipulação do banco de dados.

Requisitos:

* RF001 (Login e autenticação)
* RF002 (Gerenciamento e cadastro de reclamações)
* RNF002 (Banco de dados)
* RNF004 (Hardwares mínimos)
* RNF008 (Backend e API)

**5ª Sprint (13/05 – 27/05)**

Objetivo**:**

* Início do desenvolvimento da documentação
* Criação de relacionamentos entre tabelas
* Integração da API de Reclamações com o frontend

Atividades realizadas:

* Continuação da integração iniciada na sprint anterior.
* Finalização da comunicação entre a API e as telas desenvolvidas.
* Validação das funcionalidades do CRUD por meio da interface.
* Associação das tabelas de Reclamação com Tags e Imagens
* Criação do README do projeto

Requisitos:

* RF002 (Gerenciamento e cadastro de reclamações)
* RF009 (Geração de pontuação para reclamações)
* RNF006 (Acessibilidade)
* RNF007 (Feedback ao usuário)
* RNF008 (Backend e API)
* RNF009 (Documentação)

**6ª Sprint (28/05 – 10/06)**

Objetivos**:**

* Adicionar rota para autenticação JWT e verificação de token para acesso das API’s
* Utilizar criptografia de senha para cadastro de usuário
* Finalizar relacionamento de Reclamações com Tags e Imagens no frontend e prosseguir com o desenvolvimento das rotas das API’s de Reclamações e Usuários
* Ajustes finais nas interfaces de Reclamação, Tags e Usuários

Atividades realizadas:

* Autenticação com JWT
* Criptografia de senhas
* Aplicacação de Middleware
* Ajustes finais na UI das API’s propostas

Requisitos:

* RF001 (Login e autenticação)
* RF002 (Gerenciamento e cadastro de reclamações)
* RNF002 (Banco de dados)
* RNF003 (Responsividade)
* RNF006 (Acessibilidade)
* RNF007 (Feedback ao usuário)
* RNF008 (Backend e API)
* RNF009 (Documentação)

**7ª Sprint (10/06 – 24/06)**

Objetivos**:**

* Finalizar cadastro de usuários e visualização, edição e exclusão das reclamações feita por um usuário.
* Finalizar documentação do projeto

Atividades realizadas:

* Relatório de Sprints finalizado
* README do projeto atualizado com informações finais
* Cadastro de reclamações integrado com as tabelas Tags e Imagens
* Exclusão, edição e pesquisa de reclamações com base no usuário com login ativo
* Cadastro de um novo usuário

Requisitos:

* RF002 (Gerenciamento e cadastro de reclamações)
* RNF007 (Feedback ao usuário)
* RNF009 (Documentação)

**Entrega Final: 17/06 e 26/06**