

Documento de Requisitos para Sistema de Reclamações

1. Visão Geral

Este documento descreve os requisitos para o desenvolvimento de um sistema de gerenciamento de reclamações de usuários. O sistema permitirá que os usuários registrem reclamações e visualizem as reclamações enviadas por eles. O back-end será desenvolvido em Java com Spring Boot e o front-end será desenvolvido em React.

2. Objetivos

- **OBJ-01:** Criar uma API para receber, armazenar e exibir reclamações de usuários.
- **OBJ-02:** Desenvolver uma interface de usuário para que os usuários possam:
 - Enviar reclamações.
 - Visualizar as reclamações que eles submeteram anteriormente.

3. Requisitos Funcionais

3.1. Cadastro de Reclamações

- **RF-01:** O sistema deve permitir que o usuário envie uma nova reclamação.
- **RF-02:** Cada reclamação deve conter:
 - **RF-02.1:** CPF do usuário (associado ao usuário autenticado).
 - **RF-02.2:** Título da reclamação.
 - **RF-02.3:** Descrição da reclamação.
 - **RF-02.4:** Data de criação da reclamação.
- **RF-03:** A API deve validar os campos obrigatórios antes de salvar a reclamação no banco de dados.

3.2. Listagem de Reclamações

- **RF-04:** O sistema deve permitir que o usuário visualize uma lista de todas as reclamações submetidas por ele.
- **RF-05:** Cada item na lista deve exibir:
 - **RF-05.1:** Título da reclamação.
 - **RF-05.2:** Descrição (resumida).
 - **RF-05.3:** Data de criação.
- **RF-06:** O usuário deve poder clicar em uma reclamação para ver mais detalhes.

3.3. Autenticação e Autorização

- **RF-07:** Utilizar a biblioteca webGate.
- **RF-08:** O sistema deve utilizar userInfo para identificação do usuário.
- **RF-09:** Um usuário não pode visualizar ou editar reclamações de outros usuários.

3.4. Detalhamento de Reclamação

- **RF-10:** O usuário deve poder clicar em uma reclamação na lista para ver os detalhes completos.
- **RF-11:** A reclamação deve exibir:
 - **RF-11.1:** Título.
 - **RF-11.2:** Descrição completa.
 - **RF-11.3:** Data de criação.
 - **RF-11.4:** Status (se for implementado gerenciamento de status no futuro).

3.5. Editar Reclamação (Opcional)

- **RF-12:** O sistema pode permitir que o usuário edite as reclamações submetidas, enquanto o status for "pendente".

3.6. Excluir Reclamação (Opcional)

- **RF-13:** O usuário pode excluir suas próprias reclamações, caso ainda estejam pendentes de resposta.

4. Requisitos Não Funcionais

4.1. Segurança

- **RNF-01:** O sistema deve garantir que somente o usuário autenticado pode visualizar ou modificar suas próprias reclamações.
- **RNF-02:** O sistema deve implementar autenticação via JWT (JSON Web Token) para o controle de sessão no front-end (Utilizando biblioteca webGate com userInfo).

4.2. Performance

- **RNF-03:** A API deve ser capaz de lidar com até 1000 requisições simultâneas sem degradação significativa de desempenho.

4.3. Usabilidade

- **RNF-04:** A interface deve ser simples e intuitiva, de fácil navegação para que os usuários possam registrar e visualizar suas reclamações de maneira eficiente.

4.4. Escalabilidade

- **RNF-05:** O sistema deve ser desenvolvido de forma a permitir fácil expansão no futuro, caso seja necessário lidar com um volume maior de reclamações e usuários.

5. Tecnologias Utilizadas

5.1. Back-end

- **TEC-01:** Linguagem: Java
- **TEC-02:** Framework: Spring Boot
- **TEC-03:** Banco de Dados: MySQL/PostgreSQL (ou outro banco relacional)
- **TEC-04:** Segurança: Spring Security com autenticação JWT
- **TEC-05:** ORM: Hibernate ou JPA

5.2. Front-end

- **TEC-06:** Linguagem: JavaScript
- **TEC-07:** Framework: React.js
- **TEC-08:** Gerenciamento de Estado: Redux ou Context API (opcional)
- **TEC-09:** Comunicação com API: Axios ou Fetch API
- **TEC-10:** Autenticação: Armazenamento do token JWT no localStorage ou sessionStorage

6. Casos de Uso

Caso de Uso 1: Envio de Reclamação

- **CU-01:** O usuário acessa a página de reclamações.
- **CU-02:** O sistema valida se o usuário está autenticado.
- **CU-03:** O usuário preenche o formulário com o título e descrição da reclamação.
- **CU-04:** O sistema valida os dados inseridos.
- **CU-05:** O sistema salva a reclamação no banco de dados.
- **CU-06:** O sistema exibe uma mensagem de sucesso.

Caso de Uso 2: Visualizar Reclamações

- **CU-07:** O usuário acessa a página de "Minhas Reclamações".
- **CU-08:** O sistema exibe uma lista com as reclamações enviadas pelo usuário autenticado.
- **CU-09:** O usuário clica em uma reclamação para ver mais detalhes.
- **CU-10:** O sistema exibe os detalhes completos da reclamação.

Com esse formato, cada requisito pode ser referenciado de forma única e clara. Isso ajuda a organização do desenvolvimento e facilita o rastreamento de mudanças ou correções.