

**SERVIÇO NACIONAL DE APRENDIZAGEM COMERCIAL - SENAC**

**CURSO TÉCNICO EM DESENVOLVIMENTO DE SISTEMAS**

**PROJETO INTEGRADOR**

**REPASSA**

**Porto Alegre**

**2025**



**SERVIÇO NACIONAL DE APRENDIZAGEM COMERCIAL - SENAC**

**CURSO TÉCNICO EM DESENVOLVIMENTO DE SISTEMAS**

**PROJETO INTEGRADOR**

**REPASSA**

**MIGUEL DROSDOWSKI BRAGA**

**Porto Alegre**

**2025**

**AGRADECIMENTOS**

…

.

**1 INTRODUÇÃO**

**Sistema de Controle e Inspeção de Aeronaves**

## 

## Apresentação Geral do Projeto

Este projeto tem como objetivo desenvolver uma plataforma digital que facilite a doação e a troca de itens entre moradores de um mesmo bairro. A proposta central é oferecer um destino sustentável para objetos que já não são utilizados, incentivando o reaproveitamento e reduzindo o desperdício.

A plataforma permitirá que os usuários publiquem itens disponíveis para doação ou troca, conectando pessoas interessadas em dar uma nova utilidade a esses objetos. Todas as interações serão restritas ao âmbito local — exclusivamente entre moradores do mesmo bairro — promovendo a colaboração comunitária, fortalecendo laços sociais e eliminando a necessidade de transações financeiras.

Em resumo, trata-se de um site voltado à economia colaborativa, que visa estimular práticas sustentáveis e solidárias por meio da troca e do compartilhamento de recursos dentro das comunidades.

## Objetivos

### 2.1 Objetivo Geral

Desenvolver uma plataforma digital que facilite a doação e a troca de itens entre moradores de um mesmo bairro

### 2.2 Objetivos Específicos

* Implementar um sistema de cadastro e autenticação de usuários, garantindo a segurança e a personalização da experiência na plataforma.
* Desenvolver um mecanismo de filtro de itens para doação ou troca, com categorização, descrição e imagens, facilitando a busca por parte dos usuários.
* Restringir as interações a usuários do mesmo bairro, com base na localização cadastrada, visando fortalecer a colaboração local e eliminar a necessidade de transporte de longa distância.
* Disponibilizar uma funcionalidade para salvar publicações de interesse, permitindo que os usuários acompanhem facilmente os itens que desejam.
* Integrar um sistema de chat vinculado a cada publicação, possibilitando a comunicação direta entre usuários interessados, de forma segura e prática.

## JUSTIFICATIVA

…

## Metodologia

Para o desenvolvimento da plataforma, foi adotada a metodologia ágil Scrum, escolhida por sua flexibilidade, organização em ciclos curtos e foco na entrega contínua de valor. O processo foi estruturado em sprints semanais, com reuniões de alinhamento realizadas a cada semana para definir prioridades, revisar o progresso das tarefas e ajustar as metas de acordo com a evolução do projeto.

O GitHub foi utilizado como principal ferramenta para o controle de versões e acompanhamento do progresso do desenvolvimento, possibilitando organização, colaboração e rastreamento de mudanças no código de forma eficiente.

As etapas do projeto seguiram a seguinte ordem:

* Levantamento de requisitos e definição das funcionalidades essenciais;
* Planejamento e organização das tarefas em sprints;
* Implementação incremental das funcionalidades;
* Testes internos e ajustes com base no feedback entre os membros da equipe;
* Finalização da plataforma para apresentação.

O uso do Scrum, aliado ao versionamento via GitHub, proporcionou um processo iterativo, colaborativo e adaptável, favorecendo a qualidade e a clareza na execução do projeto.

## Arquitetura do Sistema (Modelagem)

## 5.1 Requisitos

### 5.1.1 Requisitos Funcionais

**RF01**. O sistema deve realizar autenticação do usuário.

**RF02**. O sistema deve permitir ao usuário cadastrar publicações.

**RF03**. O sistema deve permitir ao usuário filtrar itens.

**RF04**. O sistema deve permitir ao usuário salvar uma publicação.

**RF05**. O sistema deve permitir ao usuário fazer um comentário em uma publicação.

**RF06**. O sistema deve permitir ao usuário excluir uma publicação.

**RF07**. O sistema deve permitir ao usuário marcar uma publicação como finalizada.

**RF08**. O sistema deve restringir a visualização de itens apenas para usuários do mesmo bairro.

**RF09**. O sistema deve permitir ao usuário solicitar suporte.

**RF10**. O sistema deve permitir que usuários administradores removam comentários.

### 5.1.2 Requisitos Não Funcionais

**RNF 01**. A base de dados deve ser protegida, permitindo acesso apenas para usuários autorizados, garantindo a integridade e a confidencialidade das informações.

**RNF 02**. O sistema deve ser desenvolvido utilizando as tecnologias de programação JavaScript (para o front-end) e Python (para o back-end).

**RNF 03**. O sistema deve ser disponibilizado como uma aplicação web responsiva, acessível por navegadores modernos em diferentes dispositivos e sistemas operacionais, por meio de hospedagem em servidor ativo 24 horas por dia.

**RNF 04**. O sistema deve utilizar o banco de dados MySQL para armazenamento e gerenciamento das informações.

**RNF 05**. O sistema deve ser compatível com navegadores populares (como Chrome, Firefox e Edge), garantindo acessibilidade multiplataforma.

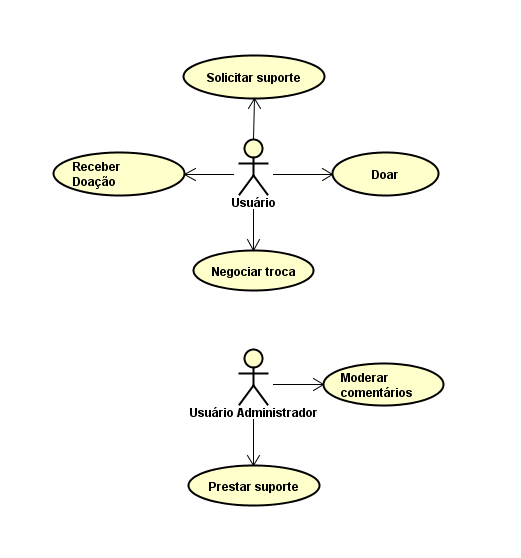
**RNF 06**. O tempo total de desenvolvimento do sistema não deve ultrapassar 24 meses, desde a fase de planejamento até a entrega final da aplicação funcional.

## 5.3 Modelagem Funcional

### Lista dos Atores

| **Ator** | **Descrição** |
| --- | --- |
| **Usuário Administrador** | O usuário administrador é responsável por manter uma estrutura básica de dados, necessário para utilização do sistema.  É responsável por gerenciar os comentários em publicações. |
| **Usuário** | O usuário pode cadastrar, editar, excluir e visualizar publicações de itens disponíveis para troca ou doação.  Além disso, o usuário pode interagir com outros usuários por meio do chat integrado às publicações, e salvar itens de interesse para consulta futura.  O acesso ao sistema é feito por meio de login com autenticação, garantindo a personalização da experiência e a segurança das informações. |

### 5.4 Diagrama Casos de Uso



### 5.4.1 Detalhamento casos de uso

Nesta seção será apresentado o detalhamento do seguintes casos de uso:

**CSU 01** – Doar

**CSU 02** – Negociar troca

**CSU 03** – Receber doação

**CSU 04** – Solicitar suporte

**CSU 05** – Moderar comentários

**CSU 06** - Prestar suporte

**CSU 01 – Doar**

| **Caso de Uso 1** | **Doar** |
| --- | --- |
| **Objetivo:** | Tem por objetivo permitir que o usuário possa cadastrar um item para doação. |
| **Ator:** | Usuário. |
| **Pré-condições:** | O usuário estar cadastrado. |
| **Cenário Principal:**   1. O usuário entra com dados para login. 2. O Sistema inicia a sessão relacionada ao usuário. 3. O usuário seleciona a opção de doar. 4. O sistema pede as informações necessárias para doar um item. 5. O usuário preenche as informações solicitadas. 6. O sistema confirma que as informações digitadas são válidas. 7. O sistema cadastra a publicação. 8. O sistema retorna uma mensagem informando que o item foi cadastrado. | |
| **Fluxo alternativo:**   1. O usuário entra com dados para login. 2. O sistema inicia a sessão relacionada ao usuário. 3. O usuário seleciona a opção de doar. 4. O sistema pede as informações necessárias para doar um item. 5. O usuário preenche as informações solicitadas. 6. O sistema confirma que as informações digitadas são inválidas. 7. O sistema exibe mensagem de erro. 8. O sistema pede que sejam preenchidas novamente as informações inválidas. | |

**CSU 02 – Negociar troca**

| **Caso de Uso 2** | **Negociar troca** |
| --- | --- |
| **Objetivo:** | Tem por objetivo a negociação de um item entre usuários. |
| **Ator:** | Usuário. |
| **Pré-condições:** | O usuário estar cadastrado. |
| **Cenário Principal:**  ***Deixado em branco para ser preenchido conforme as regras de negócios do programa são implementadas.*** | |

**CSU 03 – Receber doação**

| **Caso de Uso 3** | **Receber doação** |
| --- | --- |
| **Objetivo:** | Tem por objetivo permitir que o usuário receba uma doação. |
| **Ator:** | Usuário. |
| **Pré-condições:** | O usuário tem que estar cadastrado. |
| **Cenário Principal:**   1. O usuário entra com dados para login. 2. O sistema inicia a sessão relacionada ao usuário. 3. O usuário seleciona uma publicação. 4. O sistema exibe a publicação. 5. O usuário seleciona a opção “solicitar doação”. 6. O sistema envia uma solicitação ao autor da publicação. 7. O usuário autor da publicação aceita a solicitação. 8. O sistema verifica que a solicitação foi aceita. 9. O sistema finaliza a publicação. | |
| **Fluxo Alternativo 01:**   1. O usuário entra com dados para login. 2. O sistema inicia a sessão relacionada ao usuário. 3. O usuário seleciona uma publicação. 4. O sistema exibe a publicação. 5. O usuário seleciona a opção “solicitar doação”. 6. O sistema envia uma solicitação ao autor da publicação. 7. O usuário autor da publicação recusa a solicitação. 8. O sistema verifica que a solicitação foi recusada. 9. O sistema envia um aviso para o usuário solicitante avisando que o pedido foi recusado. | |
| **Fluxo Alternativo 02:**   1. O usuário entra com dados para login. 2. O sistema inicia a sessão relacionada ao usuário. 3. O usuário seleciona uma publicação. 4. O sistema exibe a publicação. 5. O usuário seleciona a opção “solicitar doação”. 6. O sistema envia uma solicitação ao autor da publicação. 7. O usuário autor da publicação não aceita e nem recusa a solicitação por 15 dias. 8. O sistema verifica que a solicitação não teve nenhuma alteração nos últimos 15 dias. 9. O sistema cancela o pedido. 10. O sistema envia um aviso para o usuário solicitante avisando que o tempo do pedido excedeu 15 dias. | |

**CSU 04 – Solicitar suporte**

| **Caso de Uso 4** | **Solicitar suporte** |
| --- | --- |
| **Objetivo:** | Tem por objetivo permitir que o usuário solicite suporte ao sistema. |
| **Ator:** | Usuário/Usuário administrador. |
| **Pré-condições:** | O usuário tem que estar cadastrado. |
| **Cenário Principal:**   1. O usuário entra com dados para login. 2. O sistema inicia a sessão relacionada ao usuário. 3. O usuário seleciona a opção “Solicitar suporte”. 4. O sistema disponibiliza um campo para o usuário preencher sua solicitação. 5. O usuário preenche a solicitação. 6. O usuário seleciona a opção de enviar a solicitação de suporte. 7. O sistema recebe a solicitação de suporte. 8. Um administrador do sistema responde a solicitação. 9. O usuário solicitante marca a solicitação como resolvida. 10. O sistema finaliza a solicitação. | |
| **Fluxo Alternativo 01:**   1. O usuário entra com dados para login. 2. O sistema inicia a sessão relacionada ao usuário. 3. O usuário seleciona a opção “Solicitar suporte”. 4. O sistema disponibiliza um campo para o usuário preencher sua solicitação. 5. O usuário não preenche a solicitação. 6. O usuário seleciona a opção de enviar a solicitação de suporte. 7. O sistema verifica que a caixa de pedido está vazia. 8. O sistema retorna uma mensagem de erro. | |
| **Fluxo Alternativo 02:**   1. O usuário entra com dados para login. 2. O sistema inicia a sessão relacionada ao usuário. 3. O usuário seleciona a opção “Solicitar suporte”. 4. O sistema disponibiliza um campo para o usuário preencher sua solicitação. 5. O usuário preenche a solicitação. 6. O usuário seleciona a opção de enviar a solicitação de suporte. 7. O sistema recebe a solicitação de suporte. 8. O usuário solicitante marca a solicitação como resolvida. 9. O sistema finaliza a solicitação. | |

**CSU 05 – Moderar comentários**

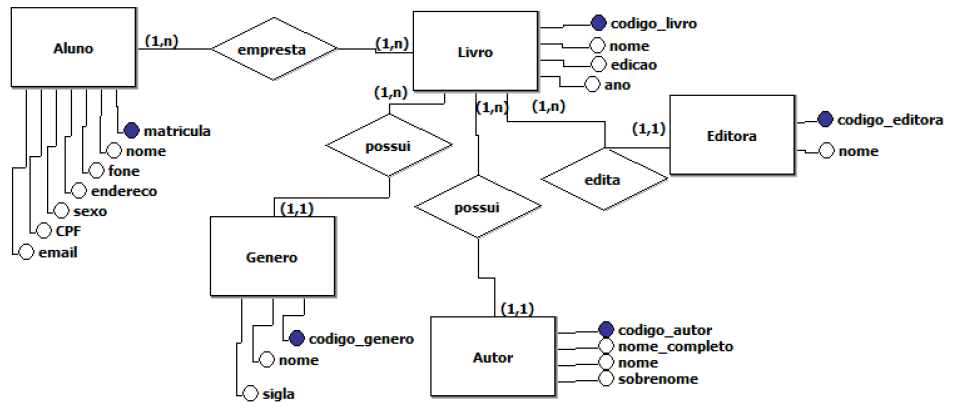
| **Caso de Uso 5** | **Moderar comentários** |
| --- | --- |
| **Objetivo:** | Tem por objetivo remover os comentários indesejados em publicações. |
| **Ator:** | Usuário Administrador. |
| **Pré-condições:** | O usuário deve estar logado com uma conta administradora. |
| **Cenário Principal:**   1. O usuário entra com dados para login. 2. O sistema inicia a sessão relacionada ao usuário. 3. O sistema verifica que o usuário logado é um administrador. 4. O sistema libera permissões de administrador ao usuário. 5. O usuário seleciona uma publicação. 6. O usuário seleciona um comentário a ser removido. 7. O sistema recebe a solicitação. 8. O sistema remove o comentário. | |

**CSU 06 – Prestar suporte**

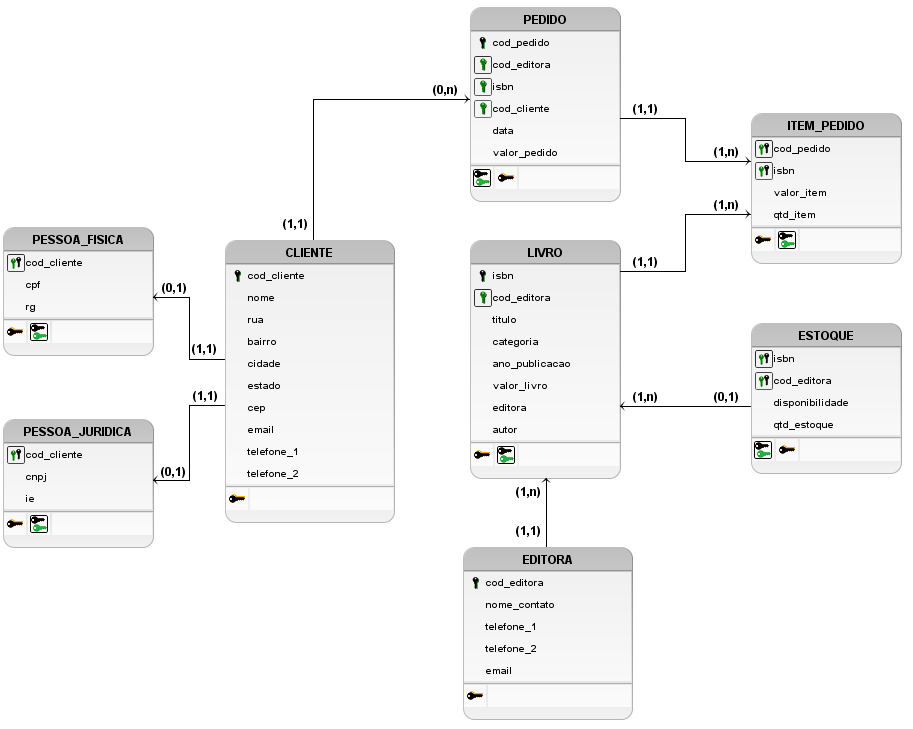
| **Caso de Uso 6** | **Prestar suporte** |
| --- | --- |
| **Objetivo:** | Tem por objetivo permitir ao usuário administrador prestar suporte aos usuários. |
| **Ator:** | Usuário Administrador. |
| **Pré-condições:** | O usuário deve estar logado com uma conta administradora. |
| **Cenário Principal:**   1. O usuário entra com dados para login. 2. O sistema inicia a sessão relacionada ao usuário. 3. O sistema verifica que o usuário logado é um administrador. 4. O sistema libera permissões de administrador ao usuário. 5. O usuário <adm> seleciona a opção de ver as solicitações em aberto. 6. O usuário <adm> responde a solicitação. 7. O sistema envia a resposta da solicitação para o usuário solicitante. 8. O usuário solicitante recebe a resposta. 9. O usuário solicitante marca a publicação como resolvida. | |
| **Fluxo Alternativo 01:**   1. O usuário entra com dados para login. 2. O sistema inicia a sessão relacionada ao usuário. 3. O sistema verifica que o usuário logado é um administrador. 4. O sistema libera permissões de administrador ao usuário. 5. O usuário <adm> seleciona a opção de ver as solicitações em aberto 6. O usuário <adm> responde a solicitação. 7. O sistema envia a resposta da solicitação para o usuário solicitante. 8. O usuário solicitante recebe a resposta. 9. O usuário solicitante contesta a resposta do administrador. 10. O sistema envia a contestação ao usuário <adm>. 11. O usuário <adm> recebe a contestação. 12. O usuário <adm> responde a contestação. 13. O sistema envia a resposta da contestação para o usuário solicitante. 14. O usuário solicitante recebe a resposta. 15. O usuário solicitante marca a publicação como resolvida. | |

1. **BANCO DE DADOS**

### 6.1 Modelo Conceitual



### 6.2 Modelo Lógico



1. **PROJETO JAVA**

### 7.1 Tutorial de instalação do Java

### 7.2 Projeto com CRUD

### 7.2.1 Telas

### 7.2.2 Conexão com Banco

### 7.3 Manual de utilização

1. **PROJETO WEB**

### 8.1 Telas HTML e CSS

### 8.2 Crud web

1. **PLANO DE TESTE**

### 8.1 Plano de Teste Desktop

### 8.2 Plano de Teste Web

1. **CONCLUSÃO**
2. **REFERÊNCIAS**