# Sistema Troca de Alimento

Versão: 1.0 Autores: Jason, Vitor e Lucas Data de Criação: 12/05/2025

## Sumário

- 1. Introdução
- 2. Tecnologias Utilizadas
- 3. Funcionalidades Principais
- 4. Estrutura do Banco de Dados
- 5. Considerações Finais
- 6. Imagens do sistema
- 7. Requisitos Funcionais
- 8. Requisitos não Funcionais
- 9. Painel de tarefas

## 1. Introdução

O Sistema Troca de Alimento é uma plataforma digital desenvolvida para facilitar a troca de alimentos entre usuários de forma prática, segura e colaborativa. O principal objetivo do sistema é promover o aproveitamento consciente dos alimentos, reduzindo o desperdício e fortalecendo o senso de comunidade.

Através do sistema, os usuários podem se cadastrar criando perfis individuais, realizar login para acessar suas funcionalidades e cadastrar alimentos que desejam disponibilizar para troca. O cadastro de alimentos inclui informações detalhadas como nome, descrição, data de validade e quantidade, garantindo transparência e segurança na troca.

Além disso, o sistema permite que os usuários avaliem as trocas realizadas, o que contribui para criar uma rede de confiança entre os participantes. Caso necessário, os usuários também podem excluir alimentos previamente cadastrados, mantendo seu perfil sempre atualizado.

A plataforma incentiva a economia colaborativa, aproximando pessoas interessadas em compartilhar recursos de maneira sustentável. Ao facilitar essas trocas, o sistema contribui diretamente para a diminuição do desperdício alimentar e para a promoção de práticas mais conscientes de consumo.

## 2. Tecnologias Utilizadas

• Front-end:

CSS

**HTML** 

Handlebars (template engine)

Back-end:

Node.js

**JavaScript** 

Banco de Dados:

MySQL

**SQLite** 

ORM para banco de dados:

Sequelize (usado para fazer a ligação entre o banco de dados e as funcionalidades do sistema, facilitando consultas, inserções e atualizações)

## 3. Funcionalidades Principais

 Cadastro de Usuário: Essa funcionalidade permite que novos usuários criem uma conta na plataforma, fornecendo informações como nome, e-mail e senha. O cadastro é essencial para controlar o acesso e permitir a personalização da experiência dentro do sistema. A senha é armazenada de forma segura, utilizando criptografia.

Objetivo:Garantir que apenas usuários registrados possam utilizar as funcionalidades de troca e gerenciar seus alimentos.

Login: Após o cadastro, o usuário pode acessar sua conta usando o e-mail e senha cadastrados.
O sistema realiza a autenticação e, em caso de sucesso, redireciona o usuário para sua área pessoal.

Objetivo: Autenticar o usuário e proteger suas informações e ações dentro da plataforma.

- Cadastro de Alimento: Os usuários autenticados podem cadastrar alimentos que desejam trocar. Cada alimento inclui campos como:
  - Nome do alimento
  - Valor estimado
  - Descrição
  - Nome do fornecedor
  - Email

#### Telefone

Essas informações são exibidas para outros usuários que estão em busca de alimentos disponíveis para troca.

Objetivo:Permitir que os usuários compartilhem alimentos com outras pessoas e promovam a redução de desperdícios.

 Avaliação das Trocas: Após uma troca ser concluída, os usuários podem avaliar a experiência, atribuindo uma nota ou deixando um comentário. Esse sistema de avaliação ajuda a manter a confiança entre os participantes e identifica usuários mais ativos e confiáveis.

Objetivo:Incentivar o bom comportamento na comunidade e promover a segurança e qualidade nas trocas realizadas.

 Exclusão de Alimento: Caso um alimento não esteja mais disponível (por ter sido trocado, vencido ou por qualquer outro motivo), o usuário pode excluí-lo do sistema. Isso ajuda a manter a base de dados atualizada, evitando que outros usuários solicitem trocas por itens que já não existem.

Objetivo: Manter as informações da plataforma sempre atualizadas e confiáveis para todos os usuários.

## 4. Estrutura do Banco de Dados

A aplicação utiliza o Sequelize como ORM (Object-Relational Mapping) para facilitar a comunicação entre o Node.js e os bancos de dados **MySQL**. O Sequelize permite modelar as tabelas como objetos JavaScript, proporcionando uma interface mais intuitiva para manipular dados.

Modelos e Relacionamentos

A base de dados é composta por três entidades principais: Usuário, Alimento e Troca.

## 5. Considerações Finais

Esta primeira versão do sistema já possui as funcionalidades essenciais para troca de alimentos entre usuários, com possibilidade de expansão para recursos adicionais, como notificações, chat e integração com serviços externos.

# 6. Imagens do sistema

#### Página inicial do sistema:



### Página de sobre:

Sistema Troca Alimentos Home Sobre Lista de Alimentos Cadastro Login

### **Sobre o Projeto**

O Sistema de Troca de Alimentos nasceu com um propósito claro: combater o desperdício e ajudar quem realmente precisa.

Todos os dias toneladas de alimentos em bom estado são descartadas enquanto milhares de pessoas passam fome. Esse sistema foi criado para mudar essa realidade – conectando quem tem alimentos sobrando com quem precisa.

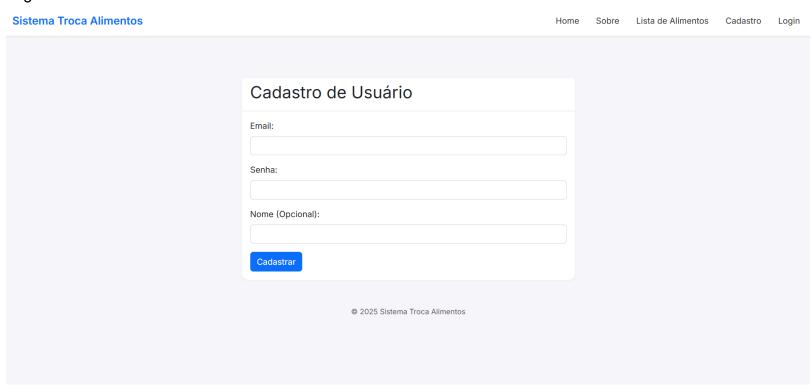
Aqui, cada doação importa. Cada cadastro pode representar uma refeição para alguém. É mais do que um sistema – é um ato de responsabilidade social, solidariedade e consciência.

A fome é real. A solução também pode ser.

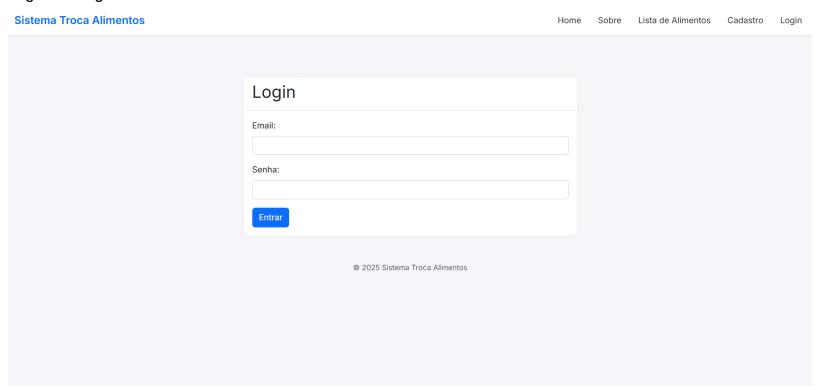
Explorar Alimentos Disponíveis

© 2025 Sistema Troca Alimentos

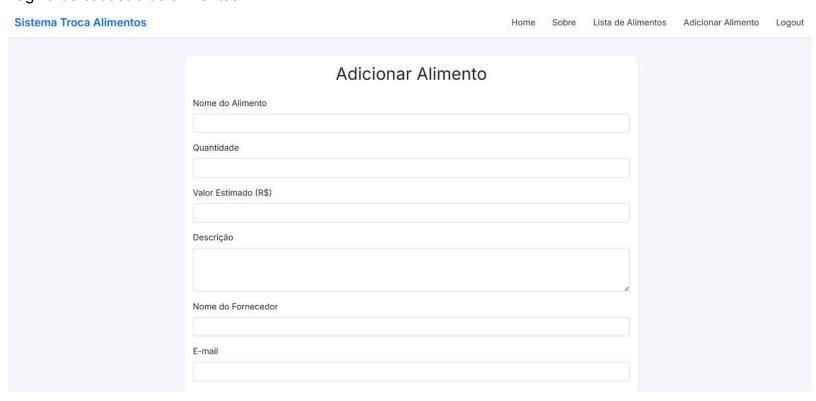
## Página de cadastro de usuário:



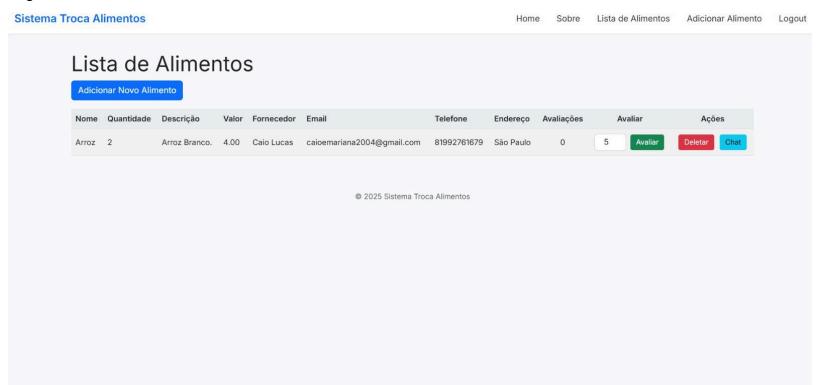
### Página de login do sistema:



### Página de cadastro de alimentos:



#### Página de lista de alimentos:



# 7. Requisitos Funcionais

## [RF001] Cadastro e Autenticação de Usuários

- O sistema deve permitir o cadastro de novos usuários.
- O sistema deve permitir que usuários façam login com e-mail e senha.
- O sistema deve validar se o e-mail é único e a senha é segura.

### [RF002] Gerenciamento de Alimentos

- O usuário autenticado deve poder cadastrar alimentos, informando:
  - Nome
  - o Descrição
  - Data de validade
- O usuário pode listar seus próprios alimentos cadastrados.
- O usuário pode editar ou excluir seus alimentos.

#### [RF003] Sistema de Troca

- O sistema deve permitir que um usuário solicite a troca de um alimento cadastrado por outro usuário.
- O dono do alimento deve poder aceitar ou recusar a solicitação.
- Uma troca aceita deve mudar seu status para "Aceita", e recusada para "Recusada".
- Após a troca, o solicitante deve poder avaliar a transação.

#### [RF004] Listagem Pública de Alimentos

- Visitantes ou usuários autenticados devem poder ver uma lista de alimentos disponíveis para troca (exceto os próprios alimentos).
- A listagem deve exibir nome, descrição, validade e quem cadastrou.

#### [RF005] Histórico de Trocas

 O usuário deve poder visualizar o histórico de trocas que participou, tanto como solicitante quanto como dono do alimento.

# 8. Requisitos não Funcionais

### [RNF001] Segurança

- As senhas dos usuários devem ser armazenadas com hash (ex: bcrypt).
- As rotas sensíveis devem ser protegidas por autenticação (middleware de sessão).
- O sistema não deve permitir ações de um usuário sobre alimentos de outro (exceto para solicitar troca).

### [RNF002] Usabilidade

- O sistema deve ter uma interface amigável e responsiva.
- Deve informar claramente ao usuário erros e confirmações (ex: login inválido, cadastro realizado com sucesso, etc).

## [RNF003] Desempenho

- O carregamento das páginas deve ser rápido mesmo com muitos alimentos cadastrados.
- As consultas ao banco devem ser otimizadas com relacionamentos bem definidos.

### [RNF004] Escalabilidade

- A estrutura deve permitir futura expansão, como:
  - Adição de imagens nos alimentos
  - Chat entre usuários
  - Sistema de notificações

### [RNF005] Portabilidade

- O sistema deve funcionar em qualquer navegador moderno e em dispositivos móveis.
- O backend deve rodar em ambientes Linux ou Windows com Node.js e MySQL/SQLite.

## 9. Painel de atividades

