

# AgroTech - API

---

## Sobre o Sistema

---

A **AgroTech API** é o back-end que sustenta uma plataforma de marketplace móvel, projetada para conectar agricultores (clientes) e fornecedores de insumos agrícolas (empresas/lojas). O sistema gerencia toda a lógica de negócio, desde o cadastro de usuários e empresas até o gerenciamento de produtos, pedidos e favoritos, com foco em segurança e escalabilidade.

## Funcionalidades Principais

---

### Gestão de Usuários (Clientes)

- Cadastro e autenticação de clientes via e-mail e senha.
- Acesso a um perfil pessoal.
- Funcionalidade para favoritar e desfavoritar empresas.

### Gestão de Empresas (Lojas/Fornecedores)

- Cadastro e autenticação de empresas.
- Gerenciamento completo do perfil da loja (dados, endereço, fotos de perfil e banner).
- Criação, atualização, exclusão e listagem de produtos.
- Gerenciamento de categorias de produtos.
- Visualização de produtos desativados.

### Gestão de Pedidos

- Criação de novos pedidos pelos clientes.
- Listagem de pedidos para clientes e empresas.
- Detalhamento de pedidos específicos.
- Funcionalidade para a empresa finalizar um pedido, atualizando seu status.

### Vitrine e Descoberta

- Listagem de categorias de produtos da plataforma.
- Exibição de banners promocionais.
- Listagem de empresas em destaque.

### Sistema de Autenticação e Segurança

- Autenticação segura baseada em **JWT (JSON Web Token)**.
- Proteção de rotas para garantir que apenas usuários autenticados acessem os recursos.
- Validação de permissões, separando as ações de clientes e empresas.

## Tecnologias Utilizadas

---

## Backend

- **Node.js:** Ambiente de execução para o JavaScript no servidor.
- **Express.js:** Framework minimalista para a construção da API e gerenciamento de rotas.
- **MySQL:** Banco de dados relacional para armazenamento dos dados.
- **JWT (JSON Web Token):** Para a criação de tokens de autenticação seguros e stateless.
- **Prisma (inferido):** ORM para facilitar a interação com o banco de dados de forma segura e tipada.
- **Bcrypt (inferido):** Para criptografia de senhas (hashing).

## Instalação e Configuração

---

### Pré-requisitos

- Node.js (versão 16 ou superior)
- MySQL
- `npm` ou `yarn`

### Passos para Instalação

1. Clone o repositório:

```
git clone [https://github.com/seu-usuario/agroplatform-api.git]
(https://github.com/seu-usuario/agroplatform-api.git)
cd agroplatform-api
```

2. Instale as dependências do backend:

```
npm install
```

3. Configure o banco de dados e as variáveis de ambiente:

```
# Crie um arquivo .env na raiz do projeto com as seguintes variáveis:
DATABASE_URL="mysql://USUARIO:SENHA@HOST:PORTA/NOME_DO_BANCO"
```

4. Execute as migrações do banco de dados para criar as tabelas:

```
npx prisma migrate dev
```

## Executando o Sistema

1. Inicie o servidor da API:

```
npm run dev
# ou
npm start
```

2. Por padrão, a API estará disponível em `http://localhost:3333` (ou a porta que você configurar).

# Rotas da API

Todas as rotas da API são prefixadas com `/api`. A maioria das rotas requer um token de autenticação.

## Autenticação e Autorização

- Todas as rotas protegidas exigem um token JWT no cabeçalho da requisição:

```
Authorization: Bearer <seu_token_jwt>
```

- ♦ **Login de Usuário:** POST `/api/usuarios/login`
- ♦ **Login de Empresa:** POST `/api/empresas/login`

## Usuários (Clientes)

Método	Rota	Descrição	Requer Auth
POST	<code>/api/usuarios</code>	Cadastra um novo usuário.	Não
GET	<code>/api/usuarios</code>	Lista todos os usuários.	Sim
GET	<code>/api/usuarios/:id_usuario</code>	Busca um usuário específico pelo ID.	Sim
GET	<code>/api/usuarios/favoritos</code>	Lista as empresas favoritas pelo usuário logado.	Sim

## Empresas (Lojas)

Método	Rota	Descrição	Requer Auth
POST	<code>/api/empresas/insereir</code>	Cadastra uma nova empresa.	Não
GET	<code>/api/empresas</code>	Lista todas as empresas.	Sim
GET	<code>/api/empresas/:id_empresa/perfil</code>	Lista o perfil de uma empresa específica.	Sim
GET	<code>/api/destaques</code>	Lista empresas em destaque na plataforma.	Sim
PUT	<code>/api/empresas/atualizar/dados/:id_empresa</code>	Atualiza os dados cadastrais da empresa.	Sim

PUT	/api/empresas/atualizar/endereco/:id_empresa	Atualiza o endereço da empresa.	Sim
<b>Método</b>	<b>Rota</b>	<b>Descrição</b>	<b>Requer Auth</b>
PUT	/api/empresas/atualizar/foto/:id_empresa	Atualiza a foto de perfil da empresa.	Sim
PUT	/api/empresas/atualizar/banner/:id_empresa	Atualiza o banner da empresa.	Sim

## Favoritos

Método	Rota	Descrição	Requer Auth
POST	/api/empresas/:id_empresa/favoritos	Usuário logado favorita uma empresa.	Sim
DELETE	/api/empresas/:id_empresa/favoritos	Usuário logado desfavorita uma empresa.	Sim

## Produtos

Método	Rota	Descrição	Requer Auth
POST	/api/empresas/:id_empresa/produto	Cadastra um novo produto para a empresa.	Sim
GET	/api/empresas/:id_empresa/produtos/:id_produto	Lista um produto específico de uma empresa.	Sim
GET	/api/empresas/desativados/:id_empresa	Lista os produtos desativados da empresa.	Sim
PUT	/api/empresas/produto/atualizar/:id_produto	Atualiza os dados de um produto.	Sim
DELETE	/api/empresas/produto/deletar/:id_produto	Exclui um produto.	Sim

## Categorias

Método	Rota	Descrição	Requer Auth
--------	------	-----------	-------------

GET	/api/categorias	Lista todas as categorias da plataforma.	Não
-----	-----------------	--	-----

Método	Rota	Descrição	Requer Auth
POST	/api/empresas/categorias/:id_empresa	Cria uma nova categoria para uma empresa.	Sim
GET	/api/empresas/categorias/:id_empresa	Lista as categorias de uma empresa específica.	Sim
DELETE	/api/empresas/categoria/deletar/:id_categoria	Exclui uma categoria da empresa.	Sim

## Pedidos

Método	Rota	Descrição	Requer Auth
POST	/api/pedidos	Cria um novo pedido para o usuário logado.	Sim
GET	/api/pedidos	Lista os pedidos do usuário logado.	Sim
GET	/api/pedidos/:id_pedido	Busca um pedido específico do usuário logado.	Sim
GET	/api/empresas/pedidos	Lista os pedidos recebidos pela empresa logada.	Sim
PUT	/api/empresas/pedidos/finalizar/:id_pedido	Marca um pedido como finalizado (para empresas).	Sim

## Banners

Método	Rota	Descrição	Requer Auth
GET	/api/banners	Lista os banners ativos da plataforma.	Sim

## Segurança

- **Autenticação:** As rotas são protegidas usando JWT. O token deve ser enviado no cabeçalho `Authorization` em todas as requisições a rotas protegidas.
- **Criptografia de Senha:** As senhas de usuários e empresas são armazenadas de forma segura no banco de dados usando algoritmos de hashing com salt (bcrypt).
- **Validação de Dados:** A API deve implementar validações na camada de controller para garantir que os dados recebidos estejam no formato correto antes de serem processados.
- **CORS:** Configurado para permitir requisições apenas de origens autorizadas (front-end da aplicação).

# AgroTech