

Documentação da API

Autenticação

Todos os endpoints (exceto /login) requerem autenticação via token JWT.

Header necessário:

Authorization: Bearer <token>

*Na pasta raiz do projeto esta disponível uma coleção exportada do Insomnia (**Insomnia_2025-06-09.yaml**), com request parametrizadas e automatizadas para facilitar os testes.

Endpoints

1. Login

- **Endpoint:** (POST) /api/auth/login
- **Descrição:** Realiza autenticação do usuário e retorna um token JWT.
- **Request Body:**

```
{
  "username": "admin",
  "password": "1213456789"
}
```

- **Resposta:**

```
{
  "token": "<jwt_token>",
  "username": "<user>",
  "role": "<role>"
}
```

- **Autenticação:** Não requerida
-

2. Buscar usuários

- **Endpoint:** (GET) /api/users
- **Descrição:** Lista todos os usuários cadastrados.
- **Resposta:**

```
{
  "data": [ {
    "id": 123,
    "username": "user",
    "password": "123456",
    "email": "user@dominio.com",
    "role": "user"
  }
]
```

- **Autenticação:** Requer token
 - **Autorização:** Apenas admin
-

3. Buscar contratos

- **Endpoint:** (GET) /api/contracts
- **Descrição:** Lista contratos registrados, com filtros opcionais por empresa e data.
- **Query Params (opcionais):**
 - empresa: Nome da empresa
 - data: Data de início (formato YYYY-MM-DD)

- **Resposta:**

```
{
  "data": [ {
    "id": 1,
    "empresa": "Empresa A",
    "data_inicio": "2024-01-01",
    "data_final": "2024-02-01"
  } ]
}
```

- **Autenticação:** Requer token
- **Autorização:** Apenas admin

Melhorias de Segurança Implementadas

- Uso de JWT com exp (expiração)
- Middleware de autenticação com jsonwebtoken
- Verificação de permissão por role
- Armazenamento seguro de JWT_SECRET via variáveis de ambiente
- Para melhorar a segurança foi alterado o backend para receber parametros de username e password, evitando sql injection;

Foi desenvolvido um frontend básico, com login e um bem vindo usuários. Para usuários **admin**, exibe alguns selects para buscar do backend os registros e exibir numa tabela. O frontend é bem simples para testar também os request. Foi usado um componente pronto de tela de login retirado de: <https://uiverse.io/mahiatlinux/thin-elephant-70>, e ajustado conforme necessário.

Guia Iniciar o Projeto

Para executar o projeto é necessário clonar o repositório e instalar os seguintes pacotes na pasta backend:

```
npm install
npm install express
npm install jsonwebtoken
npm install dotenv
npm install cors
```

```
npm install --save-dev nodemon
```

E para iniciar o backend usar o comando
`npx nodemon server.js`

Na pasta frontend:

Instalar o pacote

```
npm install -g http-server
```

E para iniciar o frontend

```
npx http-server
```