

# Documentação da API Fundação Guia

---

Esta documentação descreve os principais endpoints existentes, os tipos de retorno, estruturas de dados criadas para a API, e o modelo atual do banco de dados. Ressalta-se que esta documentação pode ser modificada conforme o desenvolvimento da API avança.

---

## Sumário

---

- [Endpoints Existentes](#)
  - [Endpoints Faltando](#)
  - [Tipos de Retorno](#)
  - [Estruturas Criadas](#)
  - [Estrutura do Banco de Dados](#)
- 

## Endpoints Existentes

---

### 1. Login do Administrador

- **POST** `/loginAdmin`
  - Realiza login do administrador.
  - Parâmetros esperados no body: `{ email, senha }`
  - Retorno: Status 200/400/401 e mensagem informando sucesso ou erro.

### 2. Notícias

- **GET** `/noticias`
  - Busca notícias filtrando por bloco, setor, exame ou quantidade de recentes.
  - Parâmetros esperados via query string: `bloco, setor, exame, recentes` (int)
  - Retorno: Lista de notícias no formato `noticia_DTO`.

### 3. Admin Actions (Exclusão de Exames)

- **DELETE** `/adminAcao/exame/:id`
  - Exclui um exame pelo ID, exige autenticação via token JWT.
  - Parâmetros: `id` via URL, token via header `Authorization`.
  - Retorno: Status, mensagem de sucesso ou erro.

### 4. Localização (em desenvolvimento)

- **GET** `/localizacao`
    - (Comentado/incompleto) Destinado a buscar informações de exames, setores ou blocos.
    - Parâmetros e retorno ainda não definidos.
- 

## Endpoints Faltando (Sugestão)

---

**Atenção:** Os endpoints sugeridos podem ser modificados conforme as necessidades do projeto.

- **POST** `/noticias`
    - Cadastro de nova notícia.
  - **PUT** `/noticias/:id`
    - Atualização de notícia existente.
  - **GET** `/localizacao/setores`
    - Listagem dos setores cadastrados.
  - **GET** `/localizacao/exames/:id`
    - Busca detalhada de exame por ID.
  - **POST** `/adminAcao/exame`
    - Cadastro de exame (com autenticação).
- 

## Tipos de Retorno

## Estrutura-padrão de retorno (ResponseBuilder, apiRetorno)

```
export interface apiRetorno<T> {  
  codigoStatus?: number;    // Código HTTP  
  mensagem?: string;        // Mensagem de erro/sucesso  
  existe?: boolean;          // Indicador de existência  
  total?: number;            // Total de itens retornados  
  body?: T;                  // Corpo da resposta  
}
```

### Exemplos de retornos específicos:

- noticiaAPIretorno: { noticias?: noticia.DTO[] }
- localizacaoAPIretorno: { exames?: exame[], setor?: setor[], bloco?: string, ... }

## Estruturas Criadas para API

### DTOs e Tipos Comuns

- admin

```
{  
  id: number;  
  nome: string;  
  email: string;  
  senha: string;  
}
```

- exame

```
{  
  nome: string;  
  descricao: string;  
}
```

- setor

```
{  
  nome: string;  
  descricao: string;  
  andar: string;  
  coordenadas: string;  
}
```

- noticia.DTO

```
{  
  titulo: string;  
  resumo: string;  
  conteudo: string;  
  data_publicacao: number;  
  tags?: string[];  
  imagens?: string[];  
  outros_links?: string[];  
}
```

- referencias

```
{  
  bloco: { imagem: string[]; descricao: string; },  
  setor: { imagem: string[]; descricao: string; }  
}
```

- params\_noticia

```
{
  recentes?: number;
  bloco?: string;
  setor?: string;
  exame?: string;
  tags?: string[];
}
```

---

## Estrutura do Banco de Dados

---

Os principais modelos referenciados são:

- **Tabela exames**
  - Campos: `id`, `nome`, `descricao`
- **Tabela setores**
  - Campos: `id`, `nome`, `descricao`, `andar`, `coordenadas`
- **Tabela admins**
  - Campos: `id`, `nome`, `email`, `senha` (armazenada com hash Argon2)
- **Tabela de notícias**
  - Campos: `id`, `titulo`, `resumo`, `conteudo`, `data_publicacao`, `tags`, `imagens`, `outros_links`

As queries utilizam o objeto `connection` para acesso via SQL (Knex.js).

---

## Observações Gerais

---

- A autenticação administrativa utiliza JWT e Argon2 para hash de senha.
  - O padrão de resposta segue o modelo `apiRetorno<T>`, padronizando status, mensagem e corpo.
  - O endpoint `/localizacao` está incompleto, sendo necessário definir rotas e retornos para blocos, setores e exames.
  - Novos endpoints podem ser definidos conforme novas necessidades e requisitos.
- 

Última atualização: 2025-10-21