

Documentação da API de PE3

Abstract—Esta documentação descreve a API de PE3, abordando as tabelas do banco de dados, rotas, repositórios, controladores, middlewares e erros. O objetivo é fornecer uma visão geral estruturada para desenvolvedores que desejam integrar e utilizar a API de maneira eficiente.

I. INTRODUÇÃO

A API de PE3 é responsável pela gestão de usuários, silos e registros de temperatura e umidade. Ela fornece endpoints para CRUD e autenticação.

II. TABELAS DO BANCO DE DADOS

A. Users

```
1 id CHAR(36) PRIMARY KEY
2 username VARCHAR(200) NOT NULL
3 password VARCHAR(250) NOT NULL
4 role ENUM('standard' | 'admin') NOT NULL
5 created_at DEFAULT CURRENT_TIMESTAMP
6 updated_at TIMESTAMP
```

B. Silos

```
1 id CHAR(36) PRIMARY KEY
2 status ENUM("active" | "inactive") NOT NULL
3 user_id CHAR(36) NOT NULL REFERENCES USERS(id) ON
  DELETE CASCADE
4 created_at DEFAULT CURRENT_TIMESTAMP
5 updated_at TIMESTAMP
```

C. Registers

```
1 id CHAR(36) PRIMARY KEY
2 temperature DECIMAL(5,2) NOT NULL
3 humidity DECIMAL(5,2) NOT NULL
4 silo_id CHAR(36) NOT NULL REFERENCES SILOS(id) ON
  DELETE CASCADE
5 created_at DEFAULT CURRENT_TIMESTAMP
```

III. ROTAS DA API

A. Registros

```
1 POST /example.url/registers -> Cria
  um registro (com silo_id no corpo)
2 GET /example.url/registers/silos/:silo_id -> Lista
  todos ou apenas o ltimo registro
3  QUERYS: ?last=true
```

B. Silos

```
1 GET /example.url/silos/:id ->
  Obt m um silo espec fico
2 PUT /example.url/silos/:id ->
  Atualiza o status de um silo espec fico
3 DELETE /example.url/silos/:id ->
  Remove um silo espec fico
4 POST /example.url/silos -> Cria
  um silo (com user_id no corpo)
5 GET /example.url/silos -> Lista
  todos os silos ou apenas os silos de um
  usu rio
6  QUERYS: ?user_id=123
```

C. Usuários

```
1 GET /example.url/users -> Lista
  todos os usu rios
2 GET /example.url/users/:id ->
  Obt m um usu rio espec fico
3 POST /example.url/users -> Cria
  um usu rio
4 POST /example.url/users/login ->
  Efetua o login de um usu rio
5 PUT /example.url/users/:id ->
  Atualiza um usu rio espec fico
6 DELETE /example.url/users/:id ->
  Remove um usu rio espec fico
```

IV. ARQUITETURA DA API

A. Repositories

```
1 # RegisterRepository
2 getAllRegisters()
3 getLastRegister()
4 createRegister()
5
6 # SiloRepository
7 getAllSilos()
8 getSilo()
9 createSilo()
10 updateStatus()
11 deleteSilo()
12
13 # UserRepository
14 getAllUsers()
15 getUser()
16 getUserByUsername()
17 createUser()
18 updateUser()
19 deleteUser()
```

B. Controllers

```
1 # RegisterController
2 index()
3 create()
4
5 # SiloController
6 show()
7 update()
8 delete()
9 create()
10 index()
11
12 # UserController
13 index()
14 show()
15 create()
16 login()
17 update()
18 delete()
```

C. Middlewares

```
1 AuthMiddleware() -> Middleware de autentica o,
  verifica e valida tokens JWT
2 ErrorMiddleware() -> Middleware de tratamento de
  erros, captura e formata respostas de erro
```

D. Errors

```
1 | HttpError -> Classe de erro usada no  
   |   |  
   |   | ErrorMiddleware
```

E. Database

```
1 | DB -> Conexão com o banco de dados, definição do  
   |   |  
   |   | Prisma Client
```

F. Server

```
1 | Arquivo de definição e inicialização do servidor
```

V. CONCLUSÃO

Esta documentação fornece um guia abrangente sobre a API de PE3, facilitando sua integração e uso por desenvolvedores. O uso de uma estrutura bem definida garante manutenibilidade e escalabilidade ao sistema.