

Gustavo de Oliveira Lima

SP3134172

## **Sistema de gerenciamento de notas**

Linguagem de Programação 3

## **Sumário**

Usando IDEs.....	3
Problemas Comuns e Soluções.....	4
Testando a API.....	5
Regras de negócio:.....	5
Login e Tokens .....	5
Controle de Acesso por Perfil.....	5
COORDENADOR:.....	5
PROFESSOR:.....	6
ALUNO: .....	6
NOTAS: .....	6
Status na matéria.....	6

## **Passos para Executar**

### **Passo 1:** Preparar o Banco de Dados

1. Instale o MySQL (se ainda não tiver)
2. Crie o banco de dados executando o script SQL fornecido no github:

[https://github.com/tavinlima/LPG3\\_gerenciador\\_de\\_notas/blob/main/Database/scripts\\_DDL\\_DML\\_DQL.sql](https://github.com/tavinlima/LPG3_gerenciador_de_notas/blob/main/Database/scripts_DDL_DML_DQL.sql)

### **Passo 2:** Configurar o Projeto

1. Abra o projeto na sua IDE (IntelliJ IDEA, Eclipse, VS Code)
2. Aguarde o Maven baixar todas as dependências
3. Configure o `application.properties` com suas credenciais do MySQL

### **Passo 3:** Compilar o Projeto

No terminal (dentro da pasta do projeto):

#### **Windows**

mvnw.cmd clean install

#### **Ou se tiver Maven instalado:**

mvn clean install

### **Passo 4:** Executar a Aplicação

#### **Windows**

mvnw.cmd spring-boot:run

#### **Ou com Maven instalado**

mvn spring-boot:run

### **Passo 5:** Verificar se está funcionando

Acesse no navegador:

- Swagger UI: <http://localhost:8080/swagger-ui.html>
- API Docs: <http://localhost:8080/api-docs>

#### **Usando IDEs**

##### **IntelliJ IDEA**

1. File → New → Project from Existing Sources

2. Selecione a pasta do projeto
3. Escolha "Maven" como tipo de projeto
4. Aguarde a indexação e download de dependências
5. Clique com botão direito em `SistemaNotasApplication.java`
6. Selecione "Run SistemaNotasApplication"

### **Eclipse**

1. File → Import → Maven → Existing Maven Projects
2. Selecione a pasta do projeto
3. Aguarde o build do projeto
4. Clique com botão direito no projeto → Run As → Spring Boot App

### **Problemas Comuns e Soluções**

**Erro:** "Package does not exist"

**\*\*Solução:\*\*** Execute `mvn clean install` para baixar dependências

**Erro:** "Port 8080 already in use"

**\*\*Solução:\*\***

1. Mate o processo na porta 8080
2. Ou adicione no application.properties: `server.port=8081`

**Erro:** "Cannot connect to database"

**\*\*Solução:\*\***

1. Verifique se MySQL está rodando
2. Confirme usuário/senha no application.properties
3. Teste conexão: `mysql -u root -p`

### **Erro de compilação do Lombok**

**\*\*Solução:\*\***

1. Instale plugin Lombok na IDE
2. IntelliJ: Settings → Plugins → Procure "Lombok"
3. Eclipse: Download lombok.jar e execute

## **Testando a API**

### **1. Fazer Login**

```
curl -X POST http://localhost:8080/api/auth/login \  
-H "Content-Type: application/json" \  
-d '{  
    "email": "joao@email.com",  
    "senha": "senha123"  
}'
```

### **2. Usar o Token Retornado**

```
curl -X GET http://localhost:8080/api/usuarios \  
-H "Authorization: Bearer SEU_TOKEN_AQUI"
```

### **3. Ou use o Swagger**

Acesse <http://localhost:8080/swagger-ui.html> e teste diretamente pela interface!

## **Postman**

Nesse arquivo encontra-se uma coleção Postman pronta para uso:

[https://github.com/tavinlima/LPG3\\_gerenciador\\_de\\_notas/blob/main/Backend/collection\\_postman.json](https://github.com/tavinlima/LPG3_gerenciador_de_notas/blob/main/Backend/collection_postman.json)

## **Regras de negócio:**

### **Login e Tokens**

- Autenticação via email e senha
- Token JWT com validade de 24 horas (86400000ms)
- Token contém: ID do usuário, email e perfil
- Toda requisição protegida precisa do header Authorization: Bearer {token}

### **Controle de Acesso por Perfil**

#### **COORDENADOR:**

- Acesso total a todos os recursos

- Único que pode criar/editar/deletar usuários
- Pode gerenciar todas disciplinas, matrículas e notas
- Acesso a todos os relatórios

**PROFESSOR:**

- Pode visualizar usuários e listas
- Pode criar disciplinas para si mesmo
- Pode editar disciplinas que leciona
- Pode lançar e editar notas dos alunos
- Pode visualizar alunos matriculados em suas disciplinas
- Acesso a relatórios de suas disciplinas

**ALUNO:**

- Pode se matricular em disciplinas
- Pode visualizar suas próprias notas
- Pode consultar seu histórico acadêmico
- Pode ver suas médias por disciplina
- NÃO pode editar notas
- NÃO pode ver dados de outros alunos

**NOTAS:**

Caso a média do aluno/estudante seja maior ou igual a 5,6 (e menor que 6), a média sera automaticamente arredondada para 6.

**Status na matéria:**

Alunos cuja média nas disciplinas seja:

- **Menor que 6 e maior ou igual a 4:** **recuperação**
- **Menor que 4:** **reprovado**
- **Maior ou igual a 6:** **aprovado**