I’M MUSIC

**Mapa do Projeto: Plataforma de Conexão Musical**

**Objetivo**

Criar uma aplicação web em Java que:

* Permite músicos se cadastrarem
* Permite visitantes pesquisarem artistas
* Armazena os dados em uma planilha do Google Sheets
* É empacotada como .jar e executada localmente

**Estrutura de Pastas**

Code

musicos-app/

├── src/

│ ├── main/

│ │ ├── java/

│ │ │ └── com/musicosapp/

│ │ │ ├── controller/

│ │ │ │ └── ArtistaController.java

│ │ │ ├── model/

│ │ │ │ └── Artista.java

│ │ │ ├── service/

│ │ │ │ └── GoogleSheetsService.java

│ │ │ └── MusicosAppApplication.java

│ │ ├── resources/

│ │ │ ├── templates/

│ │ │ │ ├── index.html

│ │ │ │ ├── cadastro.html

│ │ │ │ └── pesquisa.html

│ │ │ ├── static/

│ │ │ │ └── css/

│ │ │ └── application.properties

├── credentials.json

├── pom.xml

└── README.md

**Empacotamento com Maven**

**pom.xml básico**

xml

<project xmlns="http://maven.apache.org/POM/4.0.0" ...>

<modelVersion>4.0.0</modelVersion>

<groupId>com.musicosapp</groupId>

<artifactId>musicos-app</artifactId>

<version>1.0.0</version>

<packaging>jar</packaging>

<properties>

<java.version>17</java.version>

</properties>

<dependencies>

<!-- Spring Boot -->

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-thymeleaf</artifactId>

</dependency>

<!-- Google Sheets API -->

<dependency>

<groupId>com.google.api-client</groupId>

<artifactId>google-api-client</artifactId>

<version>1.34.1</version>

</dependency>

<dependency>

<groupId>com.google.oauth-client</groupId>

<artifactId>google-oauth-client</artifactId>

<version>1.34.1</version>

</dependency>

<dependency>

<groupId>com.google.apis</groupId>

<artifactId>google-api-services-sheets</artifactId>

<version>v4-rev20230227-2.0.0</version>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

</plugin>

</plugins>

</build>

</project>

**Etapas do Projeto**

**1. Backend Java**

* Artista.java: classe com nome, estilo, especialidade, link, bio, foto
* GoogleSheetsService.java: conecta à planilha, lê e escreve dados
* ArtistaController.java: endpoints REST para cadastro e busca

**2. Frontend HTML (Thymeleaf)**

* index.html: página inicial com botões
* cadastro.html: formulário para músicos
* pesquisa.html: busca por estilo ou nome

**3. Configuração**

* application.properties: configurações básicas
* credentials.json: credenciais da API do Google (colocar em resources)

**Como rodar e empacotar**

1. **Executar localmente**

bash

mvn spring-boot:run

1. **Empacotar como** .jar

bash

mvn clean package

java -jar target/musicos-app-1.0.0.jar

**Extras que você pode adicionar**

* Página de perfil do artista
* Filtros por estilo musical
* Autocomplete com JavaScript
* Validação de formulário

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Hospedagem gratuita online | Use o MongoDB Atlas sem pagar | | 🧩 Estrutura flexível (JSON) | Perfeito para dados variados como bio, links, fotos | | 🚀 Integração fácil com Spring Boot | Spring Boot tem suporte nativo via Spring Data MongoDB | | 🔒 Sem necessidade de cartão | Cadastro simples e seguro |   **🛠 Como adaptar seu projeto para MongoDB**  **1. Criar conta no MongoDB Atlas**   * Acesse https://www.mongodb.com/atlas * Crie uma conta gratuita * Crie um cluster gratuito (Shared Tier) * Crie um banco chamado musicos * Crie uma coleção chamada artistas * Pegue a **connection string** (ex: mongodb+srv://usuario:senha@cluster.mongodb.net/musicos)   **2. Adicionar dependência no pom.xml** |