Minhas anotações do projeto:

API Application Interface - é o que fica exposta para WEB. é a porta de entrada do seu back-end.

REST - é um padrão que especifica como uma API deve ser implementada para ser considerada REST. Seguindo os padrões do REST você terá uma API REST :

1 - Tem que ser uma aplicação cliente/servidor;

2 - Comunicação é stateless ( não guarda status, ou seja o resultado da requisição não

depende de algo que o sistema deva guardar para retornar o resultado. Ex. o retorno

não depende de algo que precisa estar armazenado previamente na sessão do usuário

para retornar )

3 - Interface uniforme com formato padronizado utilizando os verbos http( utilização

dos verbos http para realização das operações )

GET - obter

POST - gravar

PUT - alterar ( verbo idempotente significa que uma requisição realizada 1 vez

ou 10 vez o resultado será o mesmo )

DELETE - deletar

4 - Utilização de Cache

5 - Sistema em camadas

6 - Código sob demanda ( opcional )

Uma porta com maçaneta podemos atribuir a maçaneta como sendo a API ( a interface )

a porta de entrada para o seu back-end ( que podemos considerar sendo a porta )

Camadas:

- Front End

- Back End

- Controladores REST ( Os controladores são as portas de entrada/interface = API )

- Camada do Serviços ( realizada as transações )

- Camada de Acesso a Dados ( realizada as transações )

Configuração do projeto dslist ( pasta: API\_REST\_LojaDeGames )

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Dependências do projeto: pom.xml

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Como solução para resolver alguns casos de incompatibilidade do maven, devemos indicar a versão do maven que desejamos utilizar no pom.xml: ( opcional, caso de problema segue solução )

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-resources-plugin</artifactId>

<version>3.1.0</version>

</plugin>

Configurar o arquivo application.properties para ter vários perfis de ambiente: teste, homologação ...

1 – no application.properties ( perfil de teste )

#criando um perfil de teste

spring.profiles.active=${APP\_PROFILE:test}

spring.jpa.open-in-view=false

cors.origins=${CORS\_ORIGINS:http://localhost:5173,http://localhost:3000}

2 – criar um outro application.properties com o nome application-test.properties

#H2 Connection - dados de conexao com o banco de dados

spring.datasource.url=jdbc:h2:mem:testedb

spring.datasource.username=sa

spring.datasource.password=

#H2 Client - habilita o console para ser utilizado no browse

spring.h2.console.enabled=true

spring.h2.console.path=/h2-console

#Show H2 - habilita a exibicao das queries SQL no console do STS

spring.jpa.show-sql=true

spring.jpa.properties.hibernate.format\_sql=true