

Projeto

Introdução

Este projeto foi desenvolvido no âmbito da cadeira de Bases de Dados. Tem como objetivo adquirir conhecimentos e desenvolver competências relacionadas com a análise e modelação de dados. É pretendido desenvolver uma API REST, que permita interagir com o web server.

Installation Manual

A API REST foi desenvolvida em Java. O primeiro passo é abrir a linha de comandos na pasta “Projeto” fornecida, onde se encontra o ficheiro “*docker-compose-java-psql.sh*”.

De seguida, usar o seguinte comando para iniciar a base de dados e a API:

(Windows) - ***sh docker-compose-java-psql.sh***

(Linux/MacOS) - ***./docker-compose-java-psql.sh***

Para testar os endpoints, foi fornecida uma coleção de requests exportados do Postman.

User Manual

Depois de realizar o *import* da coleção de requests estes devem ser visíveis no Postman podendo assim testar todos os endpoints criados. Para realizar estes pedidos, é necessário fazer a autenticação de um utilizador para obter o seu *Authorization Token*, que é necessário para enviar alguns *requests*.

Ao realizar qualquer request que seja feito por um cliente, é necessário colocar na tab “Authorization” um *Token* previamente obtido (selecionar Bearer Token).

Ao enviar pedidos do tipo PUT e POST é necessário complementar os requests com certos detalhes, que devem ser inseridos na tab “Body”. Esta informação deve ser dada no formato JSON.

Dado que o formato de alguns pedidos não são especificados no enunciado, iremos de seguida indicar o seu formato.

- **Listar todos os leilões em que o utilizador tenha atividade**

Para listar a informação dos leilões sobre os quais um utilizador tenha interagido, é necessário fornecer o seu *userid* como parâmetro. Para o fazer, coloca-se o valor na tab “Params” -> Path Variables na coluna “VALUE” associada à chave “userid”.

- **Efetuar uma licitação num leilão**

Para efetuar uma licitação, o utilizador terá que fornecer na tab “Body” os parâmetros necessários para a sua criação: *auctionId* e o valor da sua *bid*.

- **Escrever mensagem no mural de um leilão**

Para escrever uma mensagem no mural de um leilão, é necessário especificar o *id* do respetivo leilão. Este *id* deve ser especificado na tab “Body”, juntamente com a *mensagem* a escrever.

- **Entrega imediata de notificações a utilizadores e notificação de licitação ultrapassada**

Ao ser efetuada uma bid num leilão, o utilizador, que tinha a bid mais alta previamente, recebe uma mensagem na sua “caixa de mensagens” a informá-lo que a sua bid foi ultrapassada. Todas as mensagens associadas a um utilizador podem ser obtidas neste endpoint. Para obter as mensagens na caixa de um utilizador, coloca-se o seu *userid* na tab “Params” -> Path Variables na coluna “VALUE” associada à chave “userid”.

- **Término do leilão na data, hora e minuto marcados**

O serviço que permite terminar o leilão foi implementado através de um endpoint que, ao ser feito um request, verifica a data de terminação de todos os leilões e altera o valor *expired* para true caso este já tenha terminado.

Other Information

A autenticação dos utilizadores foi implementada com o auxílio de JWT (Java JSON Web Tokens). A chave de decodificação está guardada na variável *secret*, e o método *oAuth* serve para validar os tokens. No endpoint de autenticação, é criado um JWT que contém o *username* e o *id* do utilizador. Nos endpoints que requerem autenticação, é chamado o método *oAuth* de forma a validar o token fornecido.

O método *getIdFromToken* devolve o id associado ao token.

O método *getCurrentTimeStamp* devolve um *TimeStamp* com o instante onde é chamado.

Development Plan

Ao longo do semestre fomos marcando certos horários para realizar o projeto em conjunto.

O número de horas de trabalho vai ser similar para ambos, rondando as 35.

Entity-relationship Diagram

