

# Projeto de Base de Dados

2022/2023

-Relatório-

Licenciatura em Engenharia Informática

Docente:

José Alexandre D'Abruzzo Pereira

Alunos:

André Pinto nº 2021213497  
Bernardo Portugal nº 2021238317  
Pedro Ramalho nº 2019248594

## Introdução

Este é o relatório da nossa aplicação, Odagordify e nele iremos resumir e explicar o processo e conteúdo do nosso projeto, nele iremos abordar o desenvolvimento de uma plataforma de streaming de música simplificada, que utiliza uma Base de Dados como fundamento.

Com base nos conhecimentos adquiridos, iremos explorar os principais aspetos deste projeto inovador, que tem como objetivo oferecer uma experiência única aos amantes da música. Ao simplificar o acesso e a gestão de músicas, artistas e álbuns, a nossa plataforma proporcionará aos utilizadores uma forma conveniente e intuitiva de desfrutar de uma vasta biblioteca musical.

## Manual de instalação

1. Instalar um IDE para a linguagem pretendida, no nosso caso o PyCharm e linguagem Python
2. Instalar o programa Postman, para fazer os requests e conectar o PyCharm com o PgAdmin4
3. Instalar o Postgres para trabalhar com a base de dados
4. Criar uma base de dados no Postgres
5. Descarregar as bibliotecas no PyCharm, flask (request, jsonify, response), pycopg2, jwt, flask\_jwt\_extended, datetime, logging, etc
6. Descarregar a pasta onde se encontra o programa em python
7. Correr o código PLSQL no Postgres

## Manual do utilizador

Para utilizar o Odagordify quando estiver aberto ao público, siga estas etapas: Execute o programa Python e abra o Postman, selecionando a coleção Odagordify. A partir daí, terá acesso a diversas funcionalidades, tais como autenticação de utilizador, reprodução de músicas, compra de cartões, assinaturas e criação de playlists premium.

Para começar, faça a autenticação. Dirija-se à função "Autenticação de Utilizador" e insira o seu nome de utilizador e senha no corpo do pedido no Postman. Ao enviar o pedido, o sistema verificará a existência do utilizador e, se estiver correto, gerará um token. Esse token é criado utilizando a biblioteca jwt e é utilizado para autenticar os seus pedidos subsequentes.

Com o token em mãos, poderá realizar outras operações no Odagordify. Adicione o token ao cabeçalho do pedido no Postman. No cabeçalho, adicione um campo chamado "Authorization" e defina o tipo como "Bearer Token". Cole o token no campo "Token" do cabeçalho.

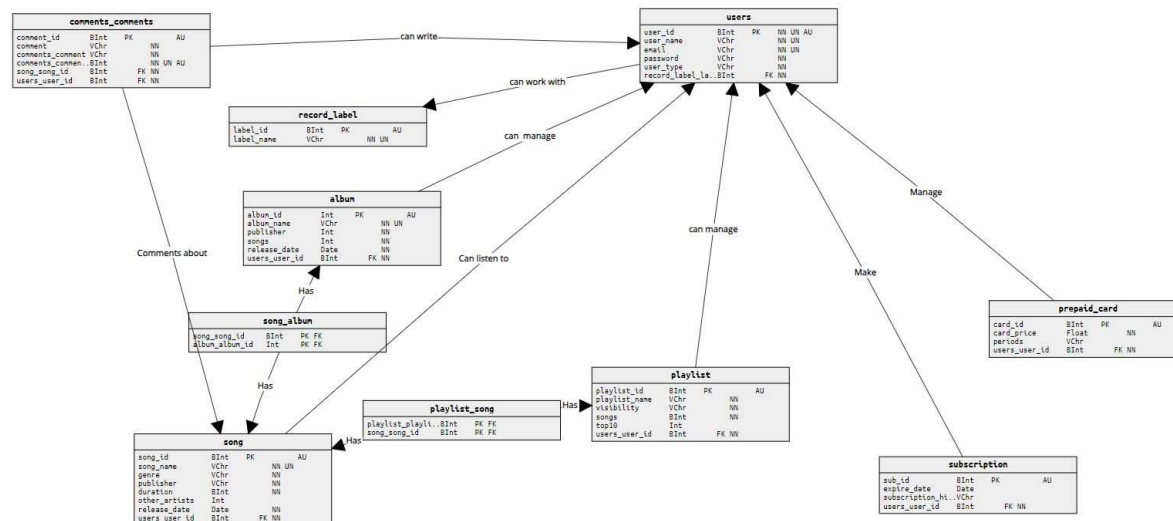
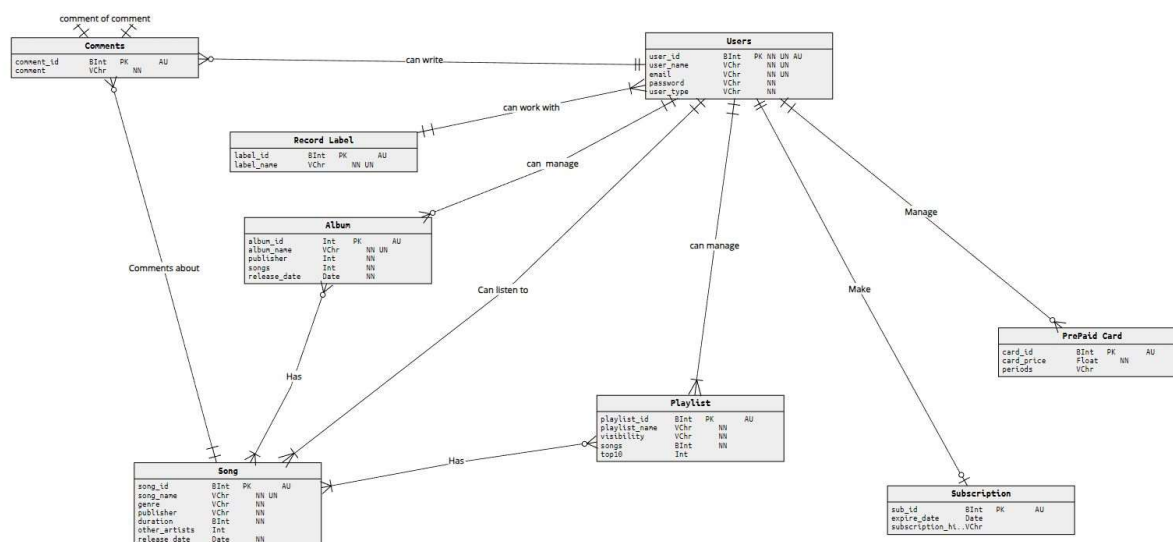
Cada operação requer parâmetros específicos, portanto, forneça as informações corretas no corpo do pedido no Postman. Por exemplo, se desejar reproduzir músicas, especifique o ID da música ou a playlist desejada. Se estiver a comprar cartões, indique o tipo de cartão e a quantidade desejada.

Certifique-se de que o cabeçalho do pedido contenha um campo "Content-Type" com o valor adequado ao tipo de pedido que está a realizar.

Após configurar corretamente o pedido no Postman, clique em "Enviar" para executar a operação desejada no Odagordify. Consulte a documentação fornecida pelo Odagordify para obter informações mais detalhadas sobre as diferentes funcionalidades e os parâmetros esperados em cada pedido.

Com estas orientações, poderá explorar as funcionalidades do Odagordify quando estiver aberto ao público e desfrutar da experiência de ouvir músicas, comprar cartões, fazer assinaturas e criar playlists premium.

# ER final e Modelos de dados relacionais



## Plano de desenvolvimento

Neste projeto, cada membro do grupo esteve envolvido durante aproximadamente 40 horas. Nesta fase do projeto, começámos por corrigir as pequenas falhas que tínhamos no ER apresentado na entrega intermédia e por rever as fichas realizadas nas aulas para consolidar funções e conhecimentos. Tal como planeado na primeira entrega, houve uma divisão equitativa de tarefas, onde cada um teve responsabilidades mais específicas, mas todos contribuíram em geral para todos os aspetos do projeto. O Pedro ficou encarregue da parte inicial, incluindo o registo de utilizador e autenticação. O Bernardo desenvolveu funções como pesquisa de músicas, comentários, adicionar música e também elaborou o relatório. O André ficou responsável por funções como adicionar álbum e subscrever a versão premium. No entanto, é difícil definir funções específicas para cada um, uma vez que trabalhámos sempre em grupo. As funções que não estão referidas foram feitas em grupo.

## Informações, decisões detalhes de design...

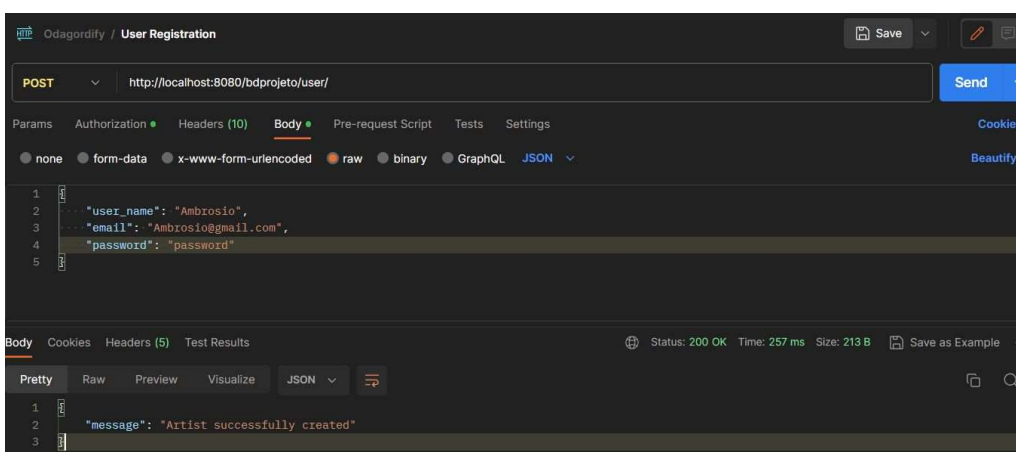
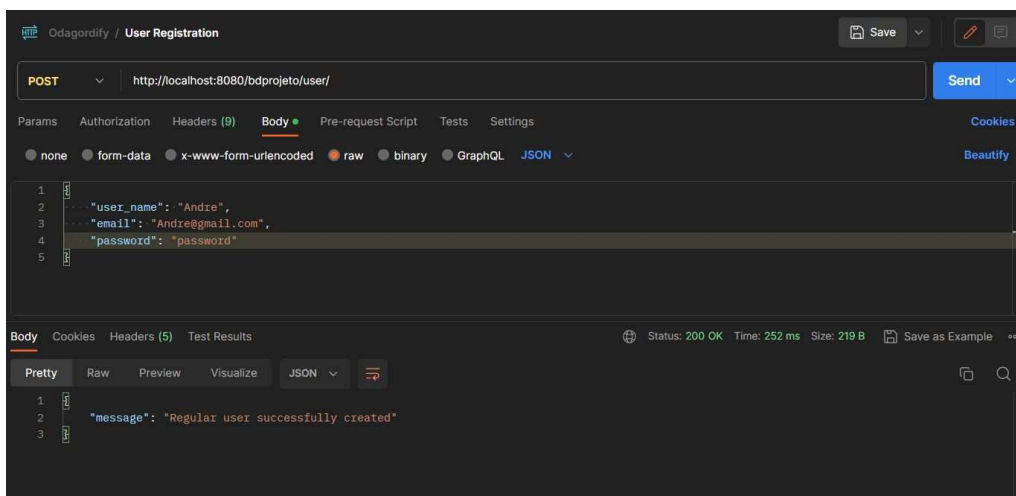
Durante o processo de desenvolvimento do projeto, priorizamos a implementação de medidas de segurança para garantir a proteção dos dados e a integridade do sistema. Para isso, utilizamos técnicas como a implementação de blocos "try" para testar novas funcionalidades com segurança e evitar erros graves.

Além disso, fizemos ajustes e modificações no diagrama ER e no tipo de algumas variáveis, adaptando-os às necessidades específicas do projeto. Também implementamos gatilhos (triggers) na base de dados para automatizar ações em resposta a eventos específicos, contribuindo para a eficiência do sistema.

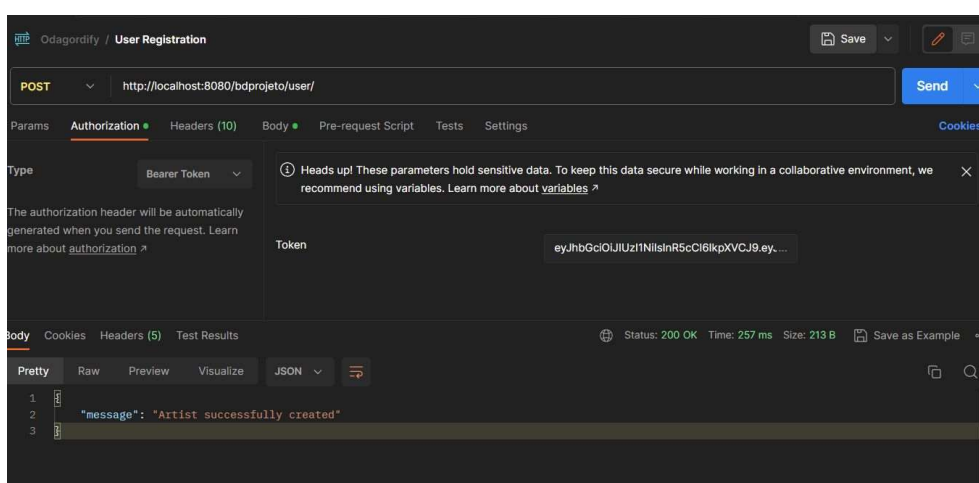
Dedicamos atenção especial à segurança dos dados, implementando uma validação rigorosa dos dados de entrada e restringindo o acesso a informações sensíveis. Essas medidas adicionais ajudaram a garantir que apenas utilizadores autorizados tenham acesso às informações e preservaram a privacidade dos utilizadores.

Ao longo do projeto, nos esforçamos para aprimorar constantemente tanto a funcionalidade quanto a segurança, adaptando-nos às necessidades em evolução e seguindo as melhores práticas de desenvolvimento de software.

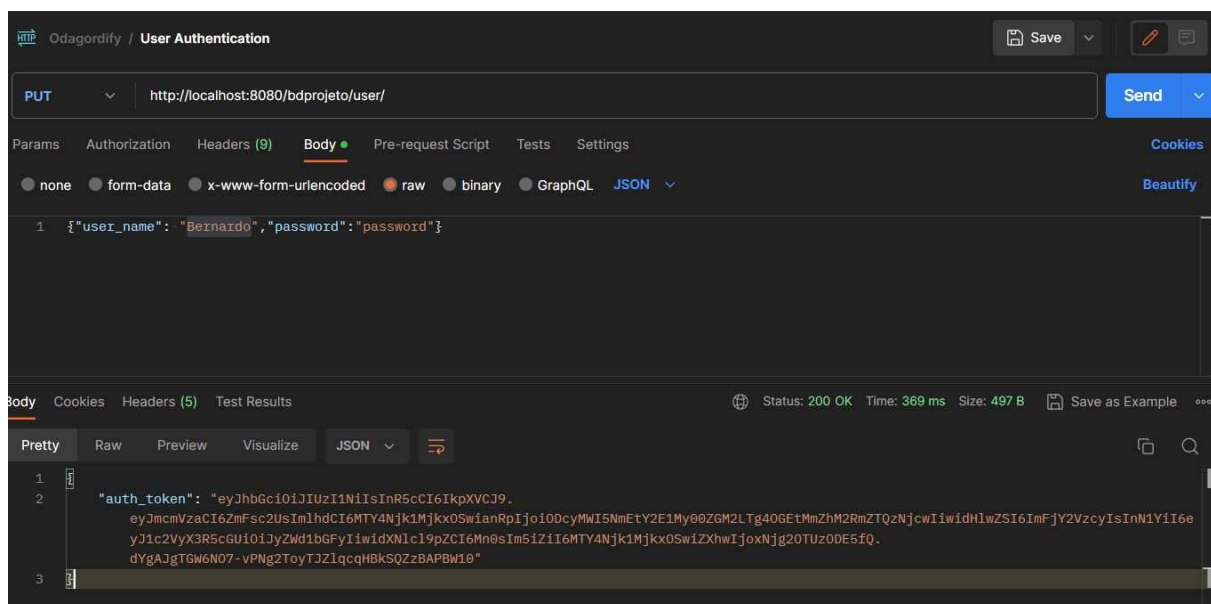
## User Registration



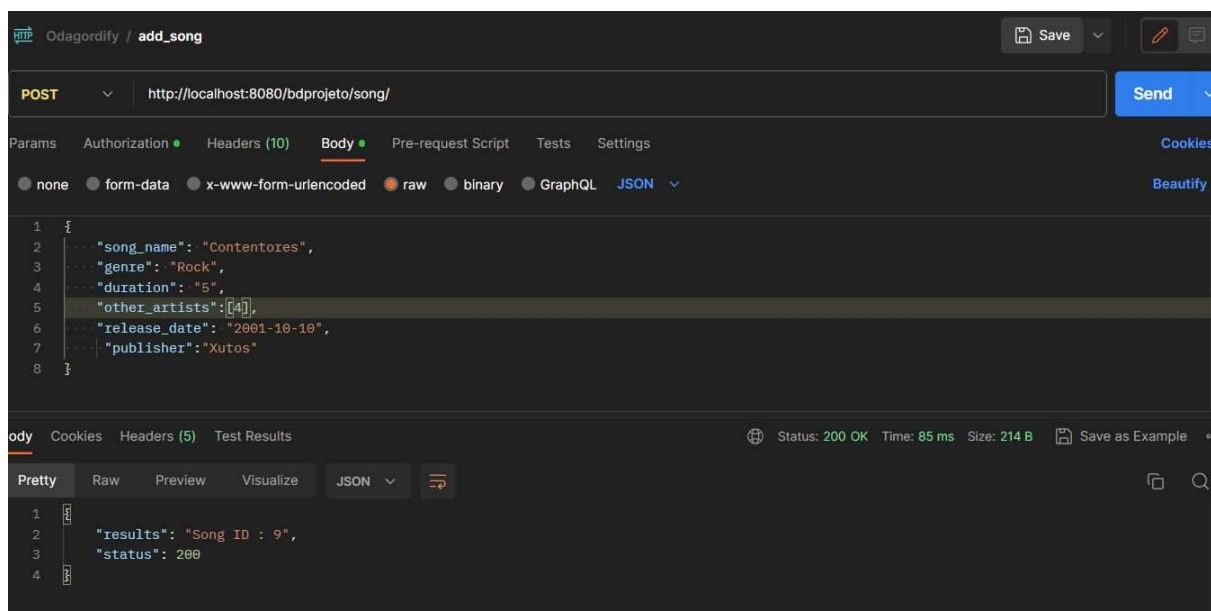
## Header com o token



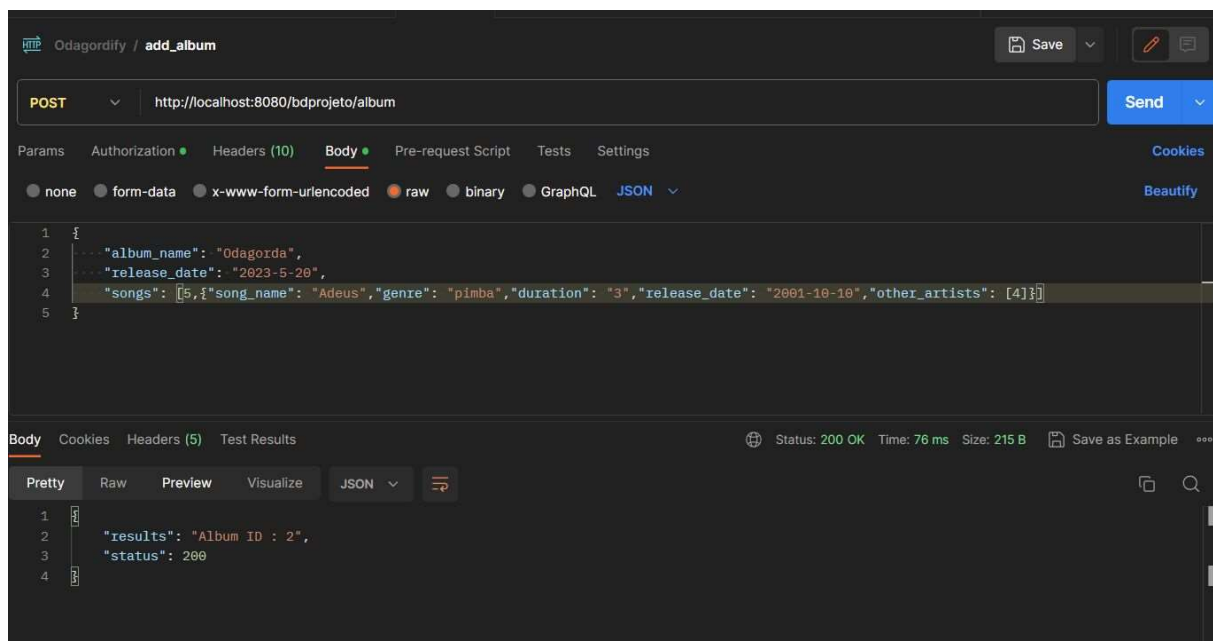
## User authentication



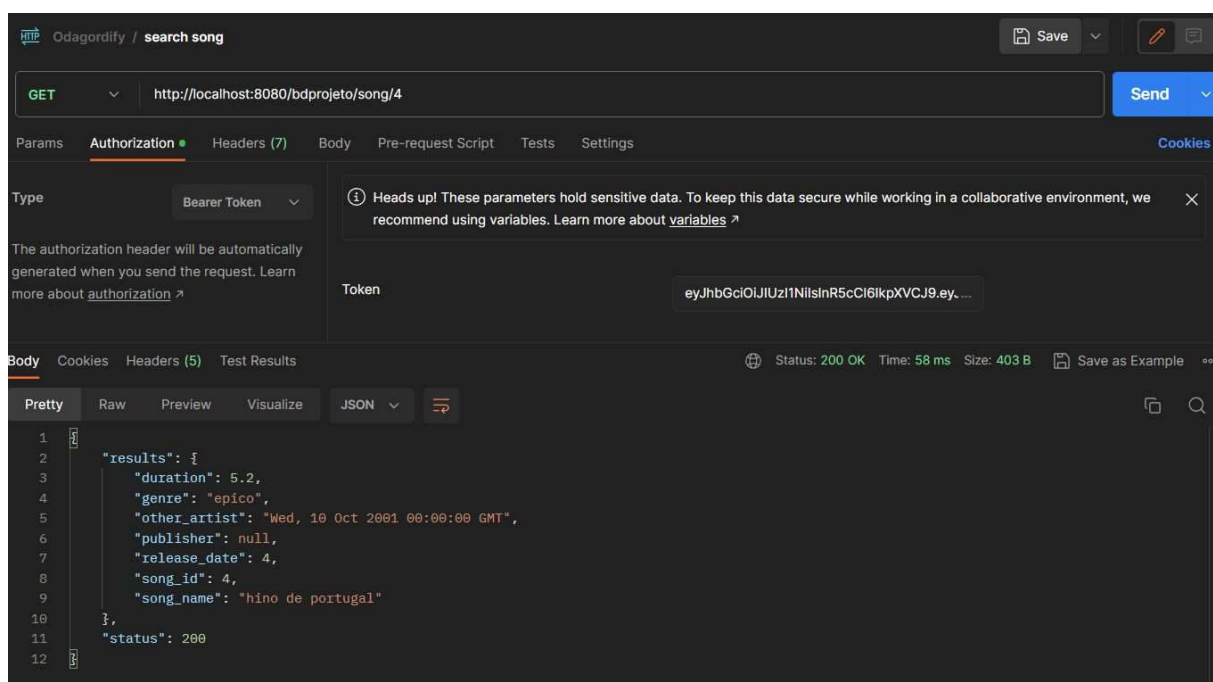
## Add song



## Add Album

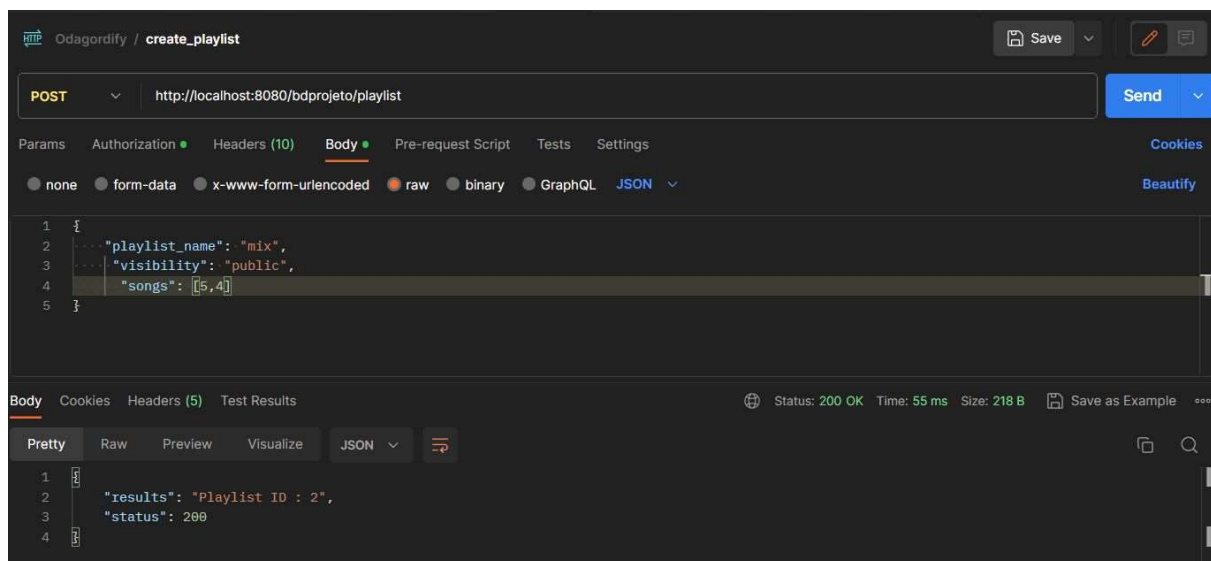


## Search song.

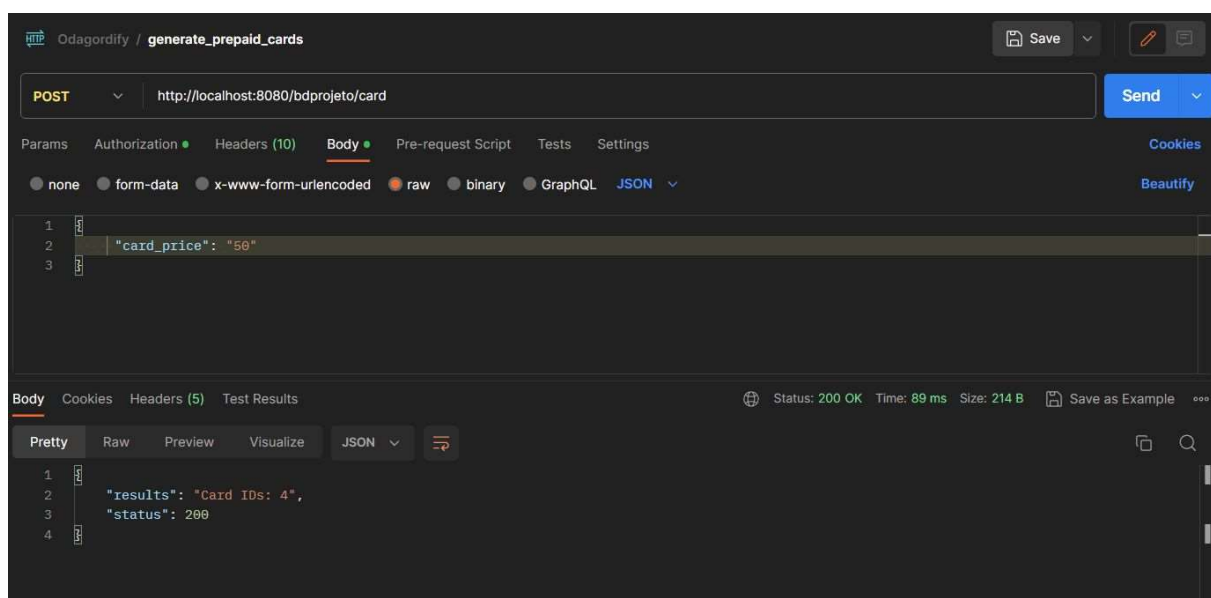




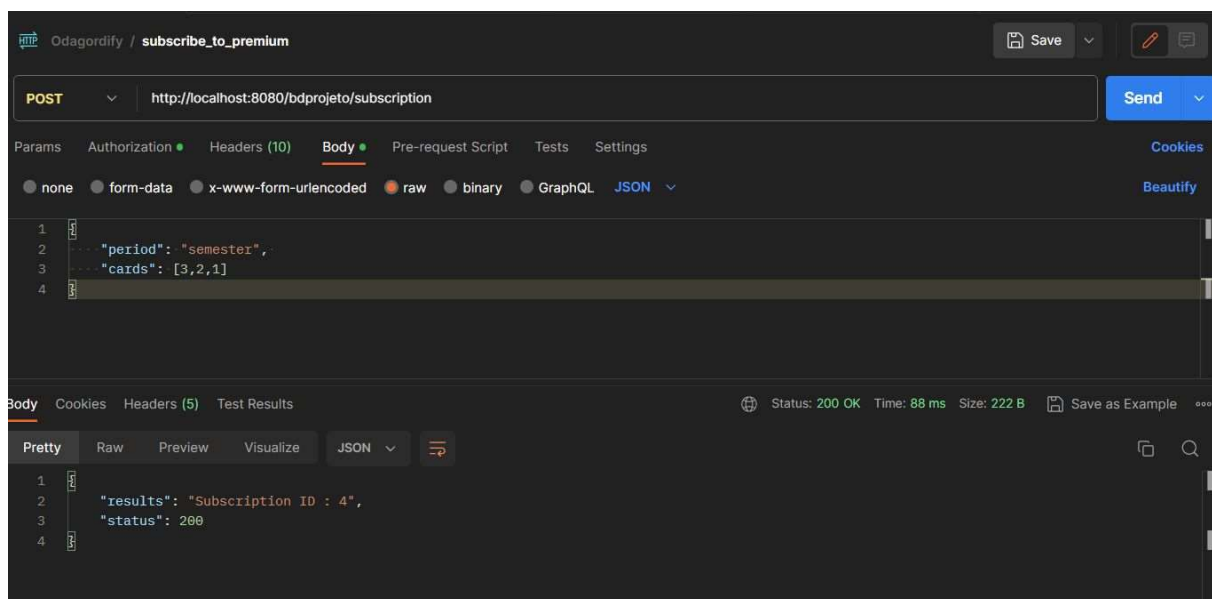
## Create playlist



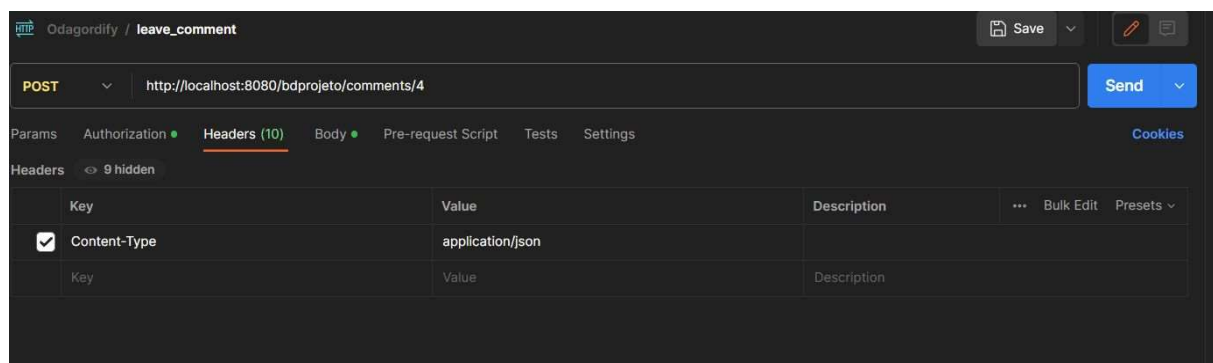
## Generate pre-paid cards



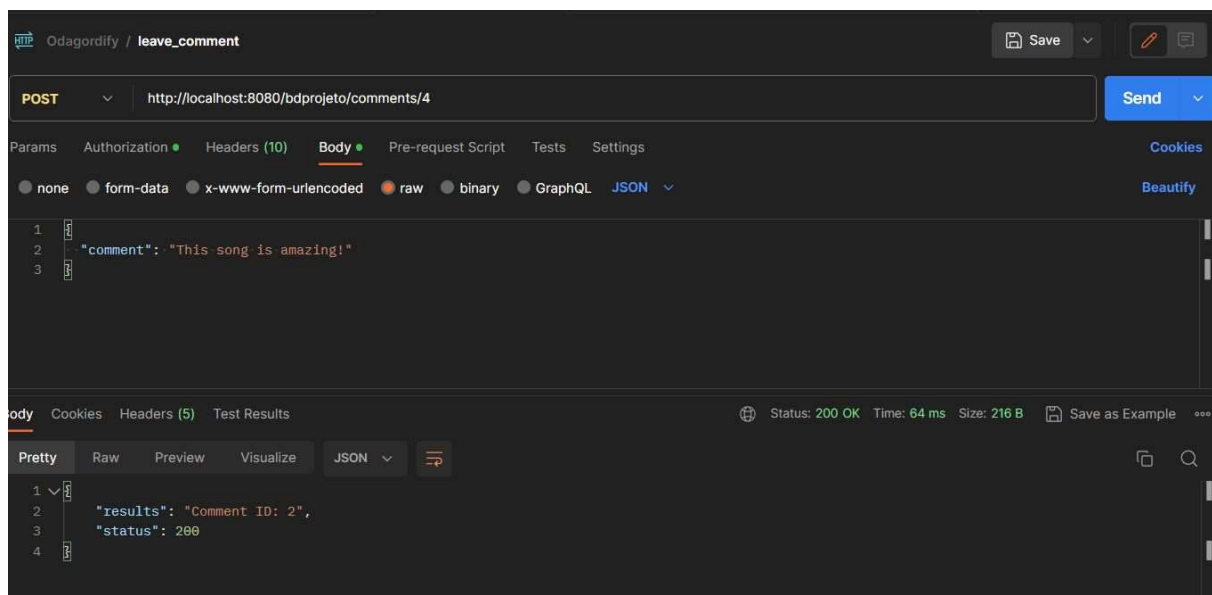
## Subscribe premium



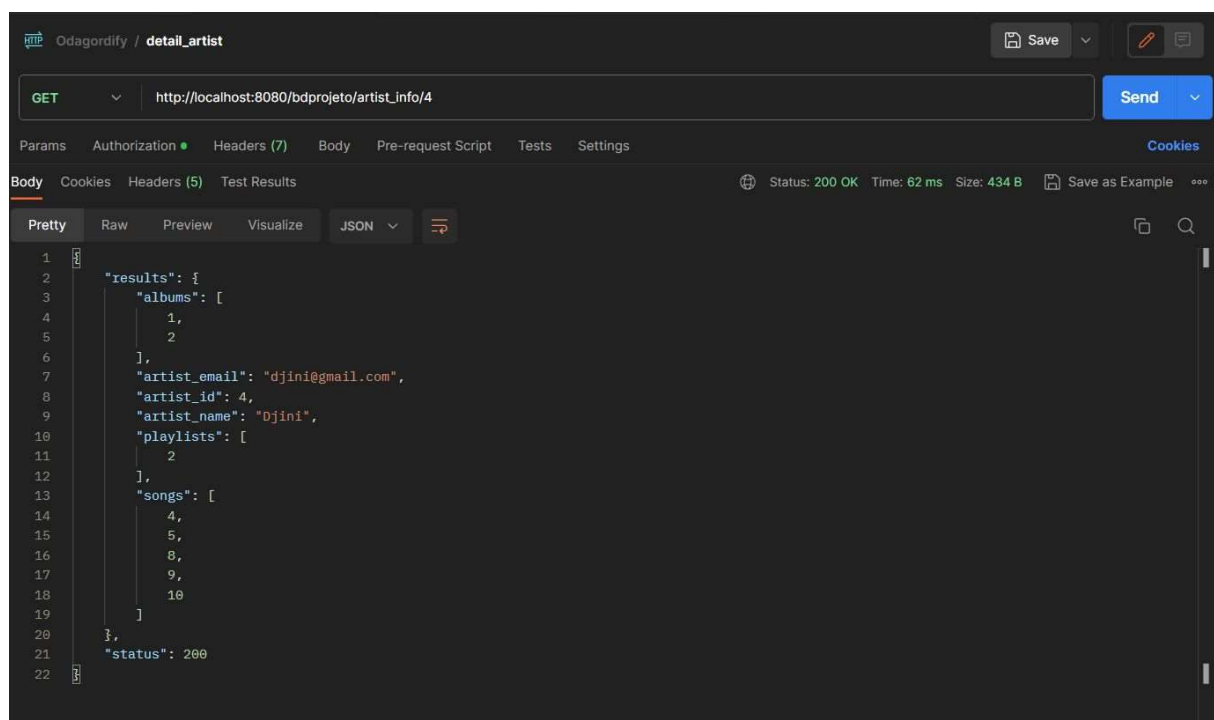
## Parâmetros de uma função



## Leave coment/feedback



## Detail artist



## Conclusão

Este projeto foi feito no âmbito da unidade curricular de Base de Dados e teve como objetivo cimentar os conhecimentos adquiridos ao longo do semestre. Depois de realizar o projeto conseguimos:

- Dominar a criação de modelos de dados conceptuais e físicos para suportar os dados da aplicação.
- Projetar, testar e implementar um sistema de base de dados.
- Instalar, configurar, gerir e afinar um sistema moderno de gestão de base de dados relacional.
- Compreender a programação do lado do cliente e do servidor em SQL e PL/pgSQL
- Aprender a usar uma REST API. Uma REST API é uma interface que permite a comunicação entre sistemas utilizando o protocolo HTTP, seguindo os princípios da arquitetura REST. No caso específico, foi utilizada uma REST API em Python no PyCharm, que se conecta ao Postman e ao pgAdmin, facilitando a criação, teste e gestão de recursos e dados.