



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

DANIEL GOMES DE LIMA

**UM DASHBOARD DE VISUALIZAÇÃO DE ESTATÍSTICAS PARA
USUÁRIOS DO SPOTIFY**

CAMPINA GRANDE - PB

2023

DANIEL GOMES DE LIMA

**UM DASHBOARD DE VISUALIZAÇÃO DE ESTATÍSTICAS PARA
USUÁRIOS DO SPOTIFY**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharel em
Ciência da Computação.**

Orientador : Professor Dr. Tiago Lima Massoni.

CAMPINA GRANDE - PB

2023

DANIEL GOMES DE LIMA

**UM DASHBOARD DE VISUALIZAÇÃO DE ESTATÍSTICAS PARA
USUÁRIOS DO SPOTIFY**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharel em
Ciência da Computação.**

BANCA EXAMINADORA:

**Professor Dr. Tiago Lima Massoni
Orientador – UASC/CEEI/UFCG**

**Professor Dr. Marcus Salerno de Aquino
Examinador – UASC/CEEI/UFCG**

**Francisco Vilar Brasileiro
Professor da Disciplina TCC – UASC/CEEI/UFCG**

Trabalho aprovado em: 28 de JUNHO de 2023.

CAMPINA GRANDE - PB

RESUMO

O streaming de áudio vem crescendo, em 2021 foram registrados mais de 500 milhões de assinantes juntando todas as plataformas, e isso tornou o ato de escutar músicas e podcasts algo mais simples e rápido. Graças a esses aplicativos, é possível ter acesso a um diverso acervo musical, de diferentes artistas e gêneros, e de podcasts. Atualmente o Spotify é o serviço de streaming de áudio mais popular no mundo com mais de 400 milhões de usuários mensais ativos. Entretanto, não é possível para o usuário consultar as estatísticas, como música mais ouvidas e métricas sobre as músicas (popularidade, duração, positividade e etc), referente ao que vem escutando ao longo dos anos, já que o Spotify disponibiliza apenas a retrospectiva anual e as músicas e artistas mais ouvidos no mês.

Por isso, esse trabalho tem o intuito de criar um dashboard acessível ao usuário com estatísticas musicais, a partir de uma investigação da API do Spotify, que é disponibilizada para desenvolvedores.

Uma contribuição adicional é, apresentar e analisar a semântica, contexto e retorno das funcionalidades da API de forma a definir os requisitos para projetar uma aplicação web com o dashboard do usuário. Este documento contém os resultados da experimentação e análise da API, além de tornar possível a utilização do dashboard com as estatísticas do usuário que estiver logado.

A STATISTICS VIEWING DASHBOARD FOR SPOTIFY USERS

ABSTRACT

Audio streaming has been growing, in 2021 more than 500 million subscribers were registered across all platforms, and this has made listening to music and podcasts simpler and faster. Thanks to these applications, it is possible to access a diverse music collection, from different artists and genres, and podcasts. Spotify is currently the most popular audio streaming service in the world with over 400 million active monthly users. However, it is not possible for the user to consult statistics, such as the most listened to songs and metrics about the songs (popularity, duration, positivity, etc.), referring to what they have been listening to over the years, since Spotify only provides the annual retrospective and the most listened to songs and artists in the month. Therefore, this work aims to create a user-accessible dashboard with music statistics, based on an investigation of the Spotify API, which is made available to developers. An additional contribution is to present and analyze the semantics, context and return of the API's functionalities in order to define the requirements to design a web application with the user's dashboard. This document contains the results of the experimentation and analysis of the API, in addition to making it possible to use the dashboard with the statistics of the user who is logged in.

Um Dashboard de Visualização de Estatísticas para Usuários do Spotify

Trabalho de Conclusão de Curso

Daniel Gomes (Aluno), Dr. Tiago Lima Massoni (Orientador)
Departamento de Sistemas e Computação
Universidade Federal de Campina Grande
Campina Grande, Paraíba - Brasil

RESUMO

O streaming de áudio vem crescendo, em 2021 foram registrados mais de 500 milhões de assinantes juntando todas as plataformas, e isso tornou o ato de escutar músicas e podcasts algo mais simples e rápido. Graças a esses aplicativos, é possível ter acesso a um diverso acervo musical, de diferentes artistas e gêneros, e de podcasts. Atualmente o Spotify é o serviço de streaming de áudio mais popular no mundo com mais de 400 milhões de usuários mensais ativos. Entretanto, não é possível para o usuário consultar as estatísticas, como música mais ouvidas e métricas sobre as músicas (popularidade, duração, positividade e etc), referente ao que vem escutando ao longo dos anos, já que o Spotify disponibiliza apenas a retrospectiva anual e as músicas e artistas mais ouvidos no mês. Por isso, esse trabalho tem o intuito de criar um dashboard acessível ao usuário com estatísticas musicais, a partir de uma investigação da API do Spotify, que é disponibilizada para desenvolvedores. Uma contribuição adicional é, apresentar e analisar a semântica, contexto e retorno das funcionalidades da API de forma a definir os requisitos para projetar uma aplicação web com o dashboard do usuário. Este documento contém os resultados da experimentação e análise da API, além de tornar possível a utilização do dashboard com as estatísticas do usuário que estiver logado.

REPOSITÓRIO

<https://github.com/dnlgomes/MyTopMusicStats>

1. INTRODUÇÃO

Música pode ser definida como uma combinação harmoniosa e expressiva de sons, bem como, a arte de se exprimir por meio de sons, seguindo regras variáveis conforme a época e a civilização [3]. Esta arte é uma das formas de manifestação cultural mais antigas, pois desde os primórdios os homens produziam formas de som, por exemplo, há indícios que na pré-história, os homens desenvolviam músicas baseadas na observação dos fenômenos da natureza [4]. Durante muito tempo, era necessário ter um músico presente, um instrumentista, para poder ouvir música. No entanto, com o advento do rádio e do disco, tornou-se possível distribuir e ouvir sons pré-gravados, o que mudou completamente nossa relação com a música. Com o surgimento e o avanço das fitas magnéticas durante a Segunda Guerra Mundial, as gravações de áudio se tornaram tão realistas que geravam dúvidas se eram realmente gravações ou orquestras tocando ao vivo, para as pessoas daquela época. Após o fim da guerra, os Estados Unidos começaram a comercializar fitas magnéticas, o que resultou no surgimento dos estúdios de gravação. Alguns anos depois, o disco de vinil foi lançado, e os artistas passaram a lançar singles e compactos com duas músicas, uma de cada lado do disco. Em

seguida, surgiu o som digital, o que levou a uma queda nas vendas de discos de vinil.

O som passou a ser representado de forma analógica e ganhou um novo formato: o CD. Com a evolução dos computadores, eles passaram a fazer parte dos processos de gravação, reduzindo os custos no processo de criação musical. Isso tornou o ato de ouvir música ainda mais acessível. Alguns anos depois, a internet começou a se popularizar e o formato de áudio MP3 surgiu como um algoritmo que reduzia drasticamente o tamanho dos arquivos de áudio, e conseqüentemente, a qualidade. No entanto, essa perda de qualidade era menos importante para os ouvintes do que a capacidade de ouvir e trocar músicas online. A indústria da música teve que se adaptar às mudanças tecnológicas, pois a música, que antes só podia ser apreciada ao vivo, hoje pode ser desfrutada em qualquer lugar, como no celular, e também pode ser totalmente produzida em um computador [5].

Hoje em dia a forma mais comum de ouvir música é a partir do streaming de áudio, que vem crescendo e tornou o ato de escutar músicas e podcasts algo muito mais simples e rápido. Graças a esses aplicativos, é possível ter acesso a um diverso acervo musical, de diferentes artistas e gêneros, e de podcasts. Em 2006, o Spotify foi lançado em meio a um cenário dominado pela pirataria na Internet. As opções legais de streaming eram caras ou limitadas. Os fundadores enxergaram na pirataria uma oportunidade para criar algo benéfico aos usuários: uma plataforma simples, com um vasto catálogo de músicas e acessível financeiramente. Além disso, o Spotify proporcionou aos artistas e gravadoras uma maneira de ganhar dinheiro e se adaptar ao mundo digital. [6]. Atualmente o Spotify é o serviço de streaming de áudio mais popular no mundo com mais de 400 milhões de usuários mensais ativos [2].

Entretanto, não é possível para o usuário consultar e visualizar detalhes estatísticos sobre sua atividade na plataforma, os detalhes que Spotify disponibiliza são a retrospectiva anual, em dezembro, com uma compilação dos dados referente a atividade do usuário durante aquele ano e as músicas e artistas mais ouvidos no mês atual sem muitos detalhes. Para o usuário é interessante que ele consiga mais detalhes sobre os seus dados no aplicativo, por mais que a retrospectiva traga bastante dados ela só ocorre uma vez por ano e os dados mensais são limitados ao mês em questão, não é possível visualizar os dados mais antigos.

Ao pesquisar sobre a API do Spotify percebe-se que através dela podemos ter acesso a mais detalhes com estatísticas do usuário. Por isso, esse trabalho tem o intuito de criar um dashboard acessível ao usuário, a partir de uma investigação da API do Spotify, que é disponibilizada para desenvolvedores, a fim de coletar as informações que ela provém para levantar e definir requisitos que resultarão na criação de um dashboard onde o

usuário possa acessar suas estatísticas, de um dado período, de forma mais detalhada.

Foi feita uma análise da semântica, contexto e retorno das funcionalidades da API, através da experimentação, que resultou na definição dos requisitos para projetar a aplicação web com o dashboard do usuário. Ao final deste documento contém os resultados da experimentação e análise da API, os requisitos do dashboard, além de ser possível a utilização do dashboard, através de uma aplicação web com as estatísticas do usuário que estiver logado.

2. SOLUÇÃO

O presente projeto tem como objetivo a criação de um dashboard onde o usuário possa acessar suas estatísticas, de um dado período, de forma mais detalhada, através da investigação da API do Spotify, que é disponibilizada para desenvolvedores, com a finalidade de coletar as informações que ela provém para levantar e definir requisitos. Desse modo será possível ao usuário logar e utilizar a aplicação web do dashboard, para ver suas estatísticas musicais.

2.1 Funcionalidades

A aplicação My Top Music Stats utiliza a API do Spotify para fornecer aos usuários uma visão abrangente de suas estatísticas musicais. Os usuários podem selecionar um período de tempo, como as últimas 4 semanas, os últimos 6 meses ou até mesmo todo o histórico de reprodução, e obter uma lista personalizada das músicas e artistas mais ouvidos nesse intervalo. Além disso, os usuários têm a flexibilidade de escolher a quantidade de músicas exibidas, variando de 10, 20 e 50 faixas em sua listagem.

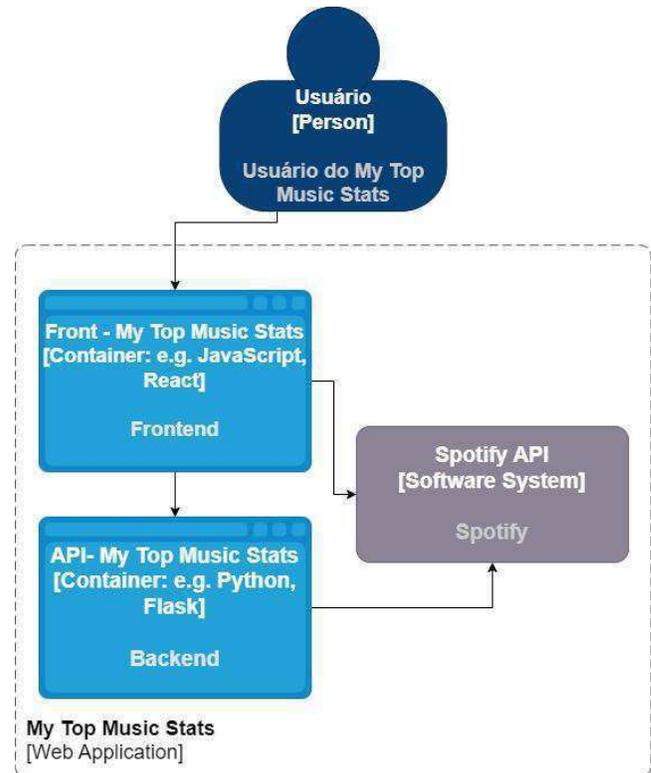
A aplicação também oferece recursos de ordenação, permitindo que os usuários classifiquem as músicas de acordo com sua popularidade, duração ou do mais ouvido ao menos ouvido. Além disso, o My Top Music Stats exibe informações adicionais sobre as músicas, como a década em que foram lançadas, seu nível de dançabilidade, energia, acústica e positividade.

3. ARQUITETURA

O sistema My Top Music Stats adotou uma arquitetura web baseada no modelo cliente-servidor [9]. Nesse modelo, o cliente faz solicitações ao servidor para obter determinados serviços, e o servidor responde fornecendo os recursos solicitados. No My Top Music Stats, o sistema é dividido em duas partes distintas. O frontend da aplicação desempenha o papel do cliente, sendo responsável por apresentar as informações de forma amigável ao usuário, utilizando os dados recebidos do servidor. Por outro lado, o backend assume o papel do servidor, aplicando manipulações nos dados.

A comunicação entre o frontend e o backend segue o modelo REST (Representational State Transfer). O modelo REST é uma abordagem arquitetural comumente usada para projetar serviços web. Ele define princípios e restrições que permitem a criação de APIs escaláveis e interoperáveis. No REST, os recursos são expostos como endpoints, e as interações ocorrem através de métodos HTTP. Essa abordagem padronizada e independente de linguagem ou plataforma tem sido amplamente adotada devido à sua simplicidade e eficiência na construção de serviços web e aplicações distribuídas [16].

A figura 1 é um diagrama de container do C4. O diagrama de container do C4 é uma técnica de modelagem arquitetural que representa a estrutura de alto nível de um sistema, mostrando os principais componentes de software (containers) e suas interações [17]. A seguir, no diagrama da figura 1, temos as relações entre o frontend (Front - My Top Music Stats), backend (API - My Top Music Stats) e a API do Spotify.



[Figura 1: Diagrama arquitetural do My Top Music Stats]

3.1 Backend

Nesta seção, descreve-se as principais decisões referente ao servidor da aplicação.

3.1.1 O backend (API - My Top Music Stats na figura 1) foi desenvolvido em Python, utilizando o framework Flask. A escolha dessa tecnologia foi baseada na facilidade de manipulação de dados e também porque Python é a mesma linguagem utilizada nos experimentos realizados no Google Colab. A API construída em Flask se comunica com a API do Spotify, permitindo a obtenção dos dados necessários para a criação das estatísticas musicais. A API é responsável pela manipulação dos dados vindo do Spotify, ou seja todos os cruzamentos de dados, cálculos de médias, moda e mediana serão feitos por essa API. Para garantir um ambiente de desenvolvimento consistente e facilitar a implantação, o Docker [15] foi utilizado durante a construção da API, permitindo a criação de contêineres para executar a aplicação de forma independente do ambiente local.

3.1.2 A API possui três endpoints principais, sendo um para o status atual da API e os outros dois para obtenção dos itens mais ouvidos, sendo um específico para músicas e outro para artistas. Esses endpoints oferecem uma experiência personalizada aos usuários, permitindo a configuração de parâmetros adicionais. Para os endpoints de músicas e artistas, é possível definir o tipo de

ordenação, que pode ser padrão, popularidade ou, no caso das músicas, duração. Além disso, é possível especificar o intervalo de tempo desejado, que pode ser as últimas 4 semanas, os últimos 6 meses ou todos os tempos. Por fim, é possível limitar a quantidade de itens retornados, variando de 1 a 50. Esses parâmetros configuráveis proporcionam aos usuários maior controle sobre as informações obtidas, adaptando a API às suas necessidades específicas de análise e descoberta musical.

3.2 Frontend

Nesta seção, descreve-se as principais decisões sobre o lado do cliente da aplicação

3.2.1 O frontend (Front - My Top Music Stats na figura 1) foi implementado utilizando o framework React [14]. Essa escolha se deve ao fato de o React ser uma biblioteca JavaScript de código aberto amplamente utilizada para a criação de interfaces de usuário.

3.2.2 A abordagem baseada em componentes do React permitiu um desenvolvimento flexível do sistema do My Top Music Stats, facilitando a reutilização de código e a criação de interfaces interativas e responsivas. Essa abordagem foi especialmente vantajosa devido à natureza do projeto, onde existem componentes similares para exibir informações sobre diferentes aspectos da música, como artistas favoritos, estatísticas de reprodução e recomendações personalizadas.

3.2.3 O controle de estado fornecido pelo React, por meio dos hooks como useState e context, foi extremamente útil na gestão e compartilhamento de dados em diferentes componentes. Isso permitiu que as informações fossem atualizadas em tempo real e que diferentes partes do aplicativo pudessem reagir de forma dinâmica às mudanças nos dados.

3.2.4 Além disso, a comunidade ativa em torno do React foi uma vantagem significativa durante o desenvolvimento do My Top Music Stats. Através de fóruns, tutoriais, bibliotecas e exemplos de código disponíveis, foi possível encontrar soluções para desafios específicos e aprender boas práticas de desenvolvimento. A contribuição da comunidade foi fundamental para resolver problemas e otimizar o código, garantindo um desenvolvimento mais eficiente e robusto.

3.2.5 Dessa forma, o React se mostrou a escolha ideal para o frontend do My Top Music Stats, proporcionando a reutilização de componentes, o controle de estado eficiente e a disponibilidade de recursos da comunidade. Essas vantagens contribuíram para a construção de uma interface intuitiva, moderna e altamente funcional para os usuários do sistema.

3.2.6 O frontend é responsável por se comunicar com a API do Spotify, para realizar o login, e com a API do My Top Music Stats para receber os dados manipulados.

4. PROCESSO DE DESENVOLVIMENTO

4.1 Investigação da api através de experimentação

Foi feita uma investigação da API que é disponibilizada pelo Spotify, através de experimentações, com a finalidade de entender como a API funciona, como é feita a conexão com aplicações, quais os endpoints da API, como os metadados que contêm as estatísticas são retornados, quais estatísticas são retornadas, e qual o período de tempo dessas estatísticas.

Foram observados mais especificamente os endpoints Get User's Top Items [7] e Get Tracks' Audio Features [8], e com isso foi percebido que são retornadas algumas informações e estatísticas sobre o que o usuário ouve e algumas características da faixa, respectivamente. Na figura 2 e 3 a seguir podemos ver os exemplos de chamada à API para o endpoint User's Top Items e Get Tracks' Audio Features respectivamente.

```
def get_top_tracks(range, token, limit):
    headers = {
        'Authorization': 'Bearer ' + token,
        'Content-Type': 'application/json'
    }

    url = "https://api.spotify.com/v1/me/top/tracks?time_range={range}&limit={limit}"

    response = requests.get(url, headers=headers)
    items = response.json().get("items")
```

[Figura 2: Chamada à User's Top Items]

```
def get_audio_features(ids, token):
    headers = {
        'Authorization': 'Bearer ' + token,
        'Content-Type': 'application/json'
    }

    url = "https://api.spotify.com/v1/audio-features?ids={ids}"

    response = requests.get(url, headers=headers)
    features = response.json().get("audio_features")
    return features
```

[Figura 3: Chamada à Get Tracks' Audio Features]

Os JSONs resultantes das figuras 2 e 3 estão nas figuras 4 e 5, respectivamente.

```
{"items":
  [
    {
      "album": {
        "album_type": "SINGLE",
        "artists": [
          {
            "external_urls": {
              "spotify":
                "https://open.spotify.com/artist/5FEcNzwbLj2pvhgZFojYAp"
            },
            "href":
              "https://api.spotify.com/v1/artists/5FEcNzwbLj2pvhgZFojYAp",
            "id": "5FEcNzwbLj2pvhgZFojYAp",
            "name": "Rap Box",
```

```

"type": "artist",
"uri": "spotify:artist:5FEcNzwbLj2pvhgZFojYAp"
},
{
"external_urls": {
"spotify":
"https://open.spotify.com/artist/4oPnjkJcLqOim9KJxvIYMz"
},
"href":
"https://api.spotify.com/v1/artists/4oPnjkJcLqOim9KJxvIYMz",
"id": "4oPnjkJcLqOim9KJxvIYMz",
"name": "Duzz",
"type": "artist",
"uri": "spotify:artist:4oPnjkJcLqOim9KJxvIYMz"
}
]
}
}

```

[Figura 4: JSON resultante do User's Top Item]

```

{"audio_features": [
{
"acousticness": 0.328,
"analysis_url":
"https://api.spotify.com/v1/audio-analysis/3VxUiSoVfw4U0f1llho5pX",
"danceability": 0.866,
"duration_ms": 210162,
"energy": 0.654,
"id": "3VxUiSoVfw4U0f1llho5pX",
"instrumentalness": 0,
"key": 8,
"liveness": 0.084,
"loudness": -7.413,
"mode": 1,
"speechiness": 0.213,
"tempo": 111.046,
"time_signature": 4,

```

```

"track_href":
"https://api.spotify.com/v1/tracks/3VxUiSoVfw4U0f1llho5pX",
"type": "audio_features",
"uri": "spotify:track:3VxUiSoVfw4U0f1llho5pX",
"valence": 0.229
}
]
}

```

[Figura 5: JSON resultante do Get Tracks' Audio Features]

A partir do experimento, que contém os responses resultantes das variações de parâmetros dos endpoints, bem como as manipulações feitas e a partir disso conseguimos definir as conclusões de que os dados disponibilizados pelo endpoint Get User's Top Items são limitantes, pois não é possível selecionar uma data específica, apenas intervalos de tempo. Isso restringe as funcionalidades do dashboard, mas ainda é possível realizar algumas manipulações com datas, popularidade e duração. O endpoint Get Tracks' Audio Features oferece características adicionais que podem ser úteis. Tais experimentações podem ser encontradas no seguinte endereço: [12].

4.1.1 Definição de requisitos da aplicação

A partir da primeira etapa, foi feita uma definição documentada das propriedades e/ou comportamentos que o dashboard deverá atender utilizando o relatório construído na etapa anterior. Com a finalidade de tornar os requisitos a base para capturar e comunicar necessidades, gerenciar expectativas, priorizar e atribuir trabalho, verificar e validar o que será possível ser feito a partir da API do Spotify e gerenciar o escopo do dashboard. A Tabela 1 apresenta os requisitos identificados para o desenvolvimento do dashboard:

| Requisito | Descrição |
|-----------|--|
| R01 | Permitir login a partir do login e logout na aplicação utilizando o Spotify. |
| R02 | Permitir seleção de limite de dados que serão retornados. |
| R03 | Permitir seleção de intervalo de tempo dos dados que serão retornados. |
| R04 | Permitir seleção do tipo de item (artista ou música) que será exibido. |
| R05 | Permitir exibição de moda, mediana e média de alguns dados. |
| R06 | Permitir exibição de um gráfico de barras com algumas informações. |
| R07 | Permitir listagem de itens. |

| | |
|-----|---|
| R08 | Para o item do tipo música, permitir a exibição das décadas mais ouvidas em um gráfico de pizza. |
| R09 | Para o item do tipo música, permitir a exibição de um gráfico de linhas com a média, moda e mediana da popularidade de acordo com as décadas. |
| R10 | Permitir seleção do tipo de ordenação da listagem. |

[Tabela 1: Requisitos do My Top Music Stats]

Através da tabela acima, é possível visualizar os requisitos estabelecidos para o dashboard, que serão considerados durante o desenvolvimento e implementação do sistema.

4.2 Prototipação de Tela

A partir dos requisitos foi feito uma tela base para exibição dessas estatísticas, foi necessário a criação de um menu de seleção para permitir que o usuário selecionasse o limite de informações que ele quer que retorne, o tipo (artista ou músicas) de informações que ele quer observar, o intervalo de tempo dos dados (R02, R03, R04) e o tipo de ordenação (R10). Para observarmos essas estatísticas precisou de uma listagem com os itens mais ouvidos (R07), tabelas para exibição de média, moda e mediana (R05), um gráfico de barras para observarmos a quantidade de álbuns, gêneros e artistas e para as características (numéricas) das músicas (R06) e, por fim, para o tipo música um gráfico de linhas (R09) e um gráfico de pizza (R08).

4.3 Implementação da API do dashboard

Com a definição dos requisitos e do relatório foi implementado a API do dashboard que se comunicará com a API do Spotify, possibilitando assim que um usuário possa logar no dashboard e ver suas estatísticas de um dado período de tempo (que terá sido definido na análise, da primeira etapa e na definição dos requisitos da segunda etapa). A implementação foi feita na linguagem Python, utilizando o framework Flask e foi utilizado o Docker, que possibilitou a criação de contêineres para executar a aplicação de maneira independente do ambiente local. Essa abordagem garantiu que a API fosse facilmente implantada e executada em diferentes ambientes. O deploy da API foi feito na AWS EC2, ficando acessível para o frontend

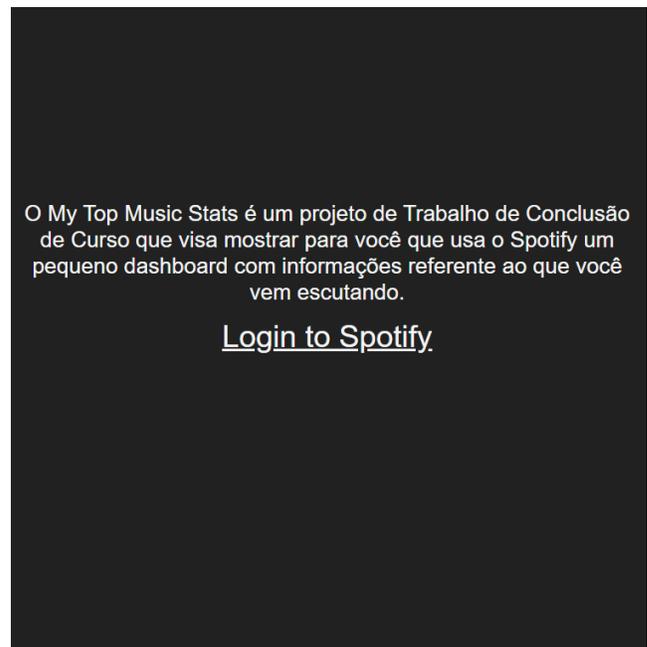
4.4 Implementação do frontend do dashboard

Por fim, foi projetado o frontend do dashboard que se comunicará com a API do dashboard. O frontend foi primeiramente projetado utilizando a ferramenta Figma, como base para o design, após isso, foi implementado em JavaScript, utilizando o framework React e a tela feita no Figma foi uma base para a implementação. Ao término da implementação e integração com a API o frontend foi implantado na Vercel, para que fosse acessível ao usuário, assim tornando possível para o usuário logar na aplicação, a partir de sua conta no Spotify, e ver suas estatísticas da forma mais detalhada possível.

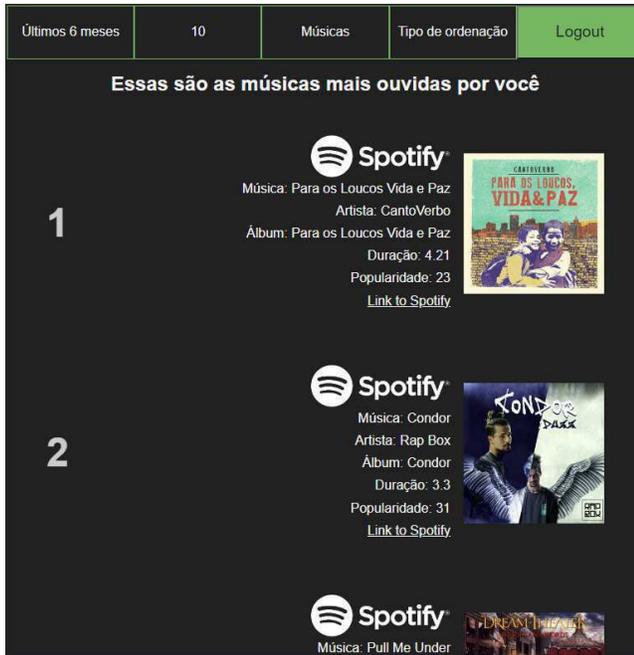
4.5 Tornando a aplicação acessível a qualquer usuário

Após a implantação do frontend, foi observado que a aplicação ainda não estava disponível para todos os usuários pois era

necessário solicitar ao Spotify uma extensão de cota, segundo a documentação [10], o modo de cota é uma configuração que determina a forma como um aplicativo pode operar, sendo dividido em dois modos distintos: modo de desenvolvimento e modo de cota estendida. Os aplicativos recém-criados iniciam no modo de desenvolvimento, o qual é ideal para aqueles em fase de construção e destinados a acessar ou gerenciar dados em uma única conta do Spotify. Já o modo de cota estendida destina-se aos aplicativos do Spotify que estão prontos para serem disponibilizados a um público mais amplo. Nessa categoria, os aplicativos podem ser instalados e/ou utilizados por um número ilimitado de usuários. Para alternar do modo de desenvolvimento para o modo de cota estendida foi necessário uma fazer uma solicitação ao Spotify, onde foi feito uma revisão que avaliou a conformidade com a política do desenvolvedor [11] e enviou o feedback por email. Para a aplicação estar em conformidade, foram feitas alterações como adicionar links e logo do Spotify e a alterações na nomenclatura da aplicação. Ao corrigir todas as pendências, a equipe do Spotify fez novamente uma revisão do aplicativo e enviou o feedback positivo no email. Com isso a aplicação ficou acessível a qualquer usuário pelo seguinte link: [13]. As figuras 6, 7 e 8 a seguir mostram um pouco do dashboard.



[Figura 6: Home do My Top Music Stats]



[Figura 7: Listagem de músicas]



[Figura 8: Tabela, Gráfico de Barras e Listagem]

A figura 6 apresenta a página inicial do My Top Music Stats, que fornece uma descrição do que é o My Top Music Stats e inclui um link para login com o Spotify. Por outro lado, as figuras 7 e 8 fornecem uma visão resumida do conteúdo da consulta de músicas mais ouvidas.

5. AVALIAÇÃO

Nesta seção, descreve-se a metodologia utilizada para a validação do sistema e os resultados obtidos.

5.1 Metodologia de Avaliação

Como forma de prova de conceito, foi passado um formulário para 18 potenciais usuários experimentarem a plataforma e fornecerem feedback sobre sua usabilidade e funcionalidade. Todos os usuários selecionados para participar dessa etapa são amantes de música e usuários ativos do Spotify, incluindo músicos e entusiastas. Eles foram recrutados por meio de uma contactação privada, levando em consideração sua afinidade com o tema.

No questionário aplicado aos participantes, foram abordados diversos aspectos relacionados ao uso da aplicação em si. Eles foram questionados se a plataforma era intuitiva, fácil de usar e se cumpria o que prometia em termos de funcionalidades. Além disso, foram indagados sobre o interesse nas estatísticas fornecidas pela plataforma e se a experiência geral havia sido agradável.

É importante ressaltar que essa etapa do projeto se trata de uma prova de conceito, não possuindo caráter de estudo científico. O objetivo principal era validar a usabilidade e funcionalidade da plataforma por meio do feedback dos usuários selecionados, buscando insights valiosos para aprimorar o desenvolvimento futuro.

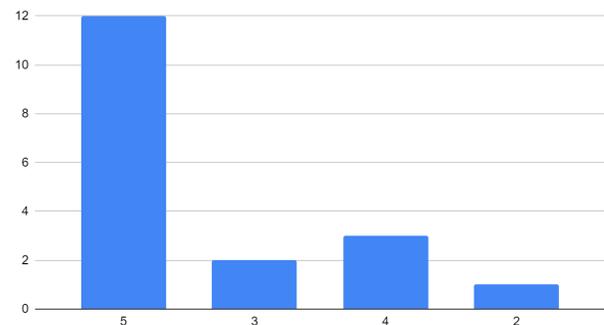
5.2 Resultados

As avaliações foram extremamente positivas, demonstrando o potencial do My Top Music Stats como uma ferramenta valiosa para os amantes da música. As respostas recebidas foram muito encorajadoras e destacaram os pontos fortes da plataforma. Aqui estão alguns destaques das avaliações:

5.2.1 Interface Intuitiva e boa experiência no uso

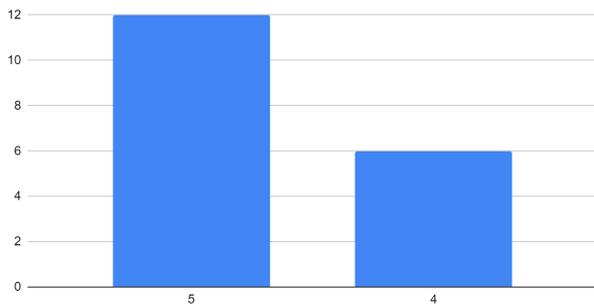
5.2.1.1 Os usuários elogiaram a interface intuitiva e amigável do My Top Music Stats. Na avaliação fica nítido que foi fácil navegar pelos menus e encontrar as informações desejadas. Também fica notório uma agradável experiência e satisfação na utilização do My Top Music Stats.

De 1 a 5 quão intuitivo é o My Top Music Stats ?



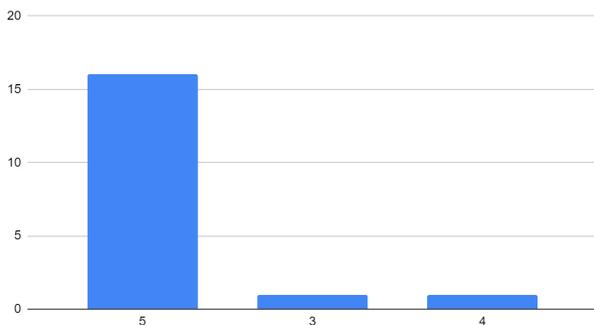
[Figura 9: Gráfico de barras referente a facilidade de uso do My Top Music Stats]

De 1 a 5 qual a sua experiência utilizando o My Top Music Stats?



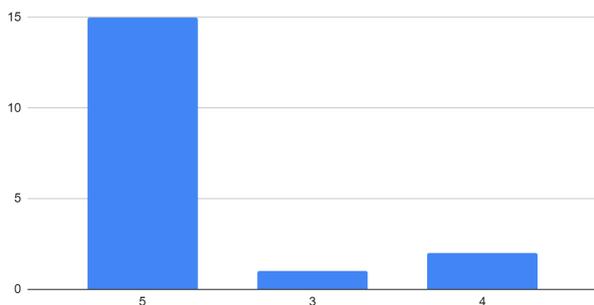
[Figura 10: Gráfico de barras referente experiência de uso do My Top Music Stats]

De 1 a 5 quão satisfeito você ficou com o My Top Music Stats?



[Figura 11: Gráfico de barras referente a satisfação no uso do My Top Music Stats]

De 1 a 5 quanto você acha que o My Top Music Stats cumpre o que promete?



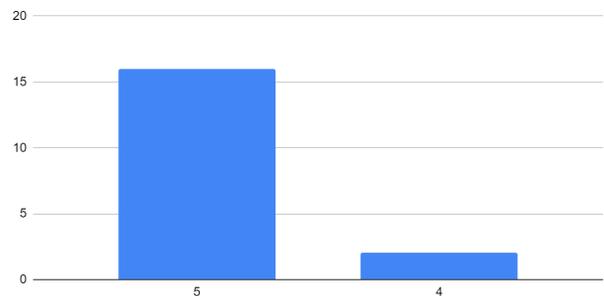
[Figura 12: Gráfico de barras que indica se o My Top Music Stats cumpre o que promete]

5.2.2 Insights Valiosos sobre Preferências Musicais

5.2.2.1 As avaliações indicam que as estatísticas fornecidas pelo My Top Music Stats deram aos usuários insights valiosos sobre suas preferências musicais. Eles apreciaram poder visualizar os artistas e músicas mais ouvidas, bem como explorar

características detalhadas sobre as faixas, como popularidade, duração, década e outros aspectos musicais.

De 1 a 5 quanto você achou interessante descobrir dados, que provavelmente você não sabia, sobre o que você vem escutando utilizando o My Top Music Stats?

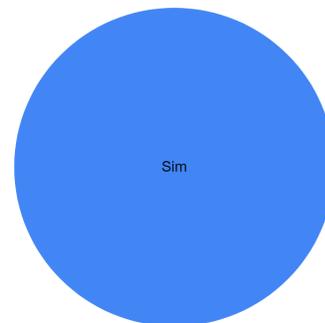


[Figura 13: Gráfico de barras referente aos insights]

5.2.3 Recomendações

5.2.3.1 A maioria dos usuários afirmou que recomendariam o My Top Music Stats a seus amigos e familiares.

Você recomendaria o My Top Music Stats?



[Figura 14: Gráfico de pizza referente a recomendação do My Top Music Stats]

5.2.4 Conclusão

Com base nas avaliações recebidas, percebe-se uma avaliação positiva à prova de conceito do My Top Music Stats. As opiniões dos usuários nos forneceram insights valiosos e direcionamento para melhorar ainda mais a plataforma e oferecer uma experiência musical ainda mais enriquecedora.

6. DISCUSSÃO E TRABALHOS FUTUROS

Embora o desenvolvimento do dashboard My Top Music Stats tenha proporcionado aos usuários uma visão de suas estatísticas musicais no Spotify, existem várias oportunidades para expandir e aprimorar ainda mais essa ferramenta. Uma área promissora para pesquisa e desenvolvimento é a criação de um sistema de recomendação personalizado que utilize as características musicais presentes no My Top Music Stats. Com base nas preferências musicais do usuário e nas características das músicas que ele mais ouve, o sistema poderia sugerir novos artistas, álbuns ou faixas que possam ser do seu interesse. Por exemplo, se um usuário escuta músicas mais enérgicas e dançáveis, o sistema iria recomendar um artista ou música, que não está presente no

histórico do usuário, que possua as mesmas características. Além disso, poderia oferecer recomendações com base em playlists ou gêneros musicais específicos, proporcionando uma experiência musical personalizada e enriquecedora.

Outro trabalho futuro interessante seria a implementação de recursos de agrupamento de músicas com base em suas características musicais. Isso permitiria aos usuários explorar diferentes agrupamentos de músicas com características semelhantes, descobrindo novos estilos musicais ou artistas que possam compartilhar semelhanças com suas preferências musicais atuais. Por exemplo, o sistema poderia agrupar músicas com alta energia e dançabilidade em uma categoria, enquanto agrupa músicas mais acústicas e instrumentais em outra. Isso proporcionaria aos usuários uma maneira intuitiva de explorar e descobrir novas músicas dentro de suas áreas de interesse.

Além disso, uma área promissora de pesquisa futura envolveria a análise de tendências musicais com base nas características presentes no My Top Music Stats. Essa análise poderia fornecer insights sobre mudanças nas preferências musicais dos usuários ao longo do tempo, identificar padrões emergentes em termos de popularidade de gêneros musicais ou até mesmo prever futuros sucessos com base nas características musicais em ascensão. Isso permitiria aos usuários acompanhar de perto as tendências musicais e explorar novos artistas ou estilos antes que se tornem amplamente conhecidos. Essas melhorias permitirão que os usuários explorem e descubram ainda mais músicas relevantes e interessantes, proporcionando uma experiência musical personalizada e agradável.

7. AGRADECIMENTOS

Gostaria de agradecer primeiramente a Deus, pois é graças a Ele que hoje estou aqui. Gostaria também de agradecer aos meus pais, irmã e à minha namorada Ana Beatriz, pelo apoio, incentivo, compreensão e amor. Sem vocês, eu não teria chegado tão longe. Meu profundo agradecimento ao meu orientador Tiago Lima Massoni, pela sua orientação dedicada, paciência e valiosos insights durante todo o processo de elaboração deste trabalho. Agradeço ao professor João Arthur Brunet, por sua colaboração na escolha do tema e pelos seus valiosos conselhos. Quero estender minha gratidão aos demais familiares e amigos que me apoiaram ao longo dessa jornada e ao curso de Computação da UFCG, por proporcionar um ambiente de aprendizado e crescimento.

8. REFERÊNCIAS

- [1] POPline. "Streaming de música: plataformas alcançam 523,9 milhões de assinantes em 2021.". Disponível em: <https://portalpopline.com.br/streaming-musica-plataformas-alcancam-523-9-milhoes-assinantes-2021/>
- [2] TudoCelular.com. "Spotify supera expectativas e registra 433 milhões de usuários mensais ativos.". Disponível em: <https://www.tudocelular.com/musica/noticias/n193370/spotify-supera-expectativas-433-milhoes-usuarios.html>
- [3] Dicionário Google. Disponível em: <https://www.google.com/search?q=defini%C3%A7%C3%A3o+de+m%C3%BAAsica&oq=defini%C3%A7%C3%A3o+de+m%C3%BAAsica&aqs=chrome..69i57j0i512l8.4508j0j7&sourceid=chrome&ie=UTF-8>
- [4] Toda Matéria. "História da Música.". Disponível em: <https://www.todamateria.com.br/historia-da-musica/>
- [5] GOU Criativa. "A evolução da Música com a inovação tecnológica.". Disponível em: <https://goucriativa.com/a-evolucao-da-musica-com-a-inovacao-tecnologica/>
- [6] Voitto. "Conheça a história do Spotify e como tudo começou!". Disponível em: <https://www.voitto.com.br/blog/artigo/spotify>
- [7] Spotify for Developers. "Get User's Top Artists and Tracks.". Disponível em: <https://developer.spotify.com/documentation/web-api/reference/get-users-top-artists-and-tracks>
- [8] Spotify for Developers. "Get Several Audio Features.". Disponível em: <https://developer.spotify.com/documentation/web-api/reference/get-several-audio-features>
- [9] Canal TI. "Arquitetura cliente-servidor.". Disponível em: <https://www.canalti.com.br/arquitetura-de-computadores/arquitetura-cliente-servidor/#:~:text=Deini%C3%A7%C3%A3o,dos%20dados%20e%20clientes%>
- [10] Spotify for Developers. "Spotify Web API Quota Modes.". Disponível em: <https://developer.spotify.com/documentation/web-api/concepts/quota-modes>
- [11] Spotify for Developers. "Spotify Developer Policy.". Disponível em: <https://developer.spotify.com/policy>
- [12] Daniel Gomes, "Analisando os endpoint Get User's Top Items e Get Tracks' Audio Features". Disponível em: <https://colab.research.google.com/drive/1hWdQzAgQfzDQUhsHWO6sYQjWxXJCyGjT?usp=sharing#scrollTo=19661mZVIxjB>
- [13] Daniel Gomes, "My Top Music Stats". Disponível em: <https://my-top-music-stats.vercel.app/>
- [14] React, "Quick Start - React". Disponível em: <https://react.dev/learn>
- [15] Docker, "Overview | Docker Documentation". Disponível em: <https://docs.docker.com/get-started/>
- [16] Fielding, R. T. (2000). Architectural Styles and the Design of Network-based Software Architectures. Disponível em: https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm.
- [17] Simon Brown, "The C4 model for visualising software architecture". Disponível em: <https://c4model.com/>