

MAC0499 - Desenvolvimento da Plataforma Spotunes

26 de Maio, 2021

Aluno: Édio Cerati Neto

Supervisor: Antônio Deusany de Carvalho Júnior

1 Introdução e Motivação

1.1 Um desafio a ser solucionado

No final da década de 90 e começo dos anos 2000, surgiram diversos dispositivos portáteis, capazes de reproduzir arquivos de música digital. Junto com a popularização de formatos de compressão de áudio, como o MP3, usuários podiam levar milhares de músicas em seus bolsos, em um dispositivo menor que uma fita cassete ou CD, que armazenavam, em média, de 10 a 15 músicas. Portanto, foi inevitável que tocadores de MP3 ganhassem a preferências dos usuários para o consumo de músicas.

Já no início da década de 2010, com o surgimento de smartphones e acesso mais fácil à internet móvel, o streaming ganha força pela praticidade. Como agora os arquivos de música estão centralizados em um servidor que provê tal serviço, o usuário passa a ter acesso a milhões de músicas facilmente, bastando escolhê-las. Diante da quantidade colossal de escolhas, surge o desafio que este trabalho procura solucionar: **como organizar e facilitar o acesso a uma coleção de milhões de arquivos de música?**

A tentativa de solucionar tal desafio se dará com o desenvolvimento de uma plataforma web chamada Spotunes, onde o usuário poderá adicionar músicas provenientes de serviços de streaming em sua biblioteca virtual, alterando seus metadados como preferir. Ele também terá acesso a dados sobre a reprodução das músicas, assim como a capacidade de criar playlists manuais ou auto-atualizáveis com base em critérios que ele definir. Por fim, a descoberta de novas músicas será recomendada com base nos metadados de sua biblioteca atual, junto com o histórico de reprodução. Mais detalhes sobre cada componente serão explorados nas próximas seções.

1.2 Bibliotecas de Música Digital

Programas como o *iTunes* e *Windows Media Player* criam uma biblioteca virtual a partir dos arquivos de música do usuário. Eles são capazes de organizar as músicas digitais a partir de seus metadados, permitindo buscar uma música específica por título, álbum, artista, gênero, dentre várias outras informações. Para auxiliar na organização, o utilizador também pode editar esses metadados como bem entender.

Neste trabalho, iremos focar nas características do *iTunes* [1], pois ele oferece uma vasta gama de opções ao usuário, incluindo o armazenamento de informações de reprodução de cada faixa musical. Alguns exemplos das informações armazenadas:

- Quantidade de reproduções;
- Data da última reprodução;
- Data de adição na biblioteca;
- Classificação de uma a cinco estrela.

1.3 Playlists

No desenvolvimento do Spotunes, haverá três tipos de playlists:

- Playlists criadas manualmente;
- Playlists Inteligentes aos moldes do iTunes [2];
- Playlists geradas automaticamente com base na biblioteca do usuário.

1.3.1 Playlists Manuais

Playlists manuais são listas de músicas em que o usuário manualmente define quais músicas devem participar e em qual ordem. Geralmente estão ligadas a questões subjetivas do usuário. Um exemplo seria: “Minhas 10 músicas favoritas”, na qual o usuário pode definir a playlist na ordem adequada para tais 10 músicas.

1.3.2 Playlists Inteligentes

Playlists Inteligentes são um recurso implementado pelo iTunes desde suas primeiras versões. O usuário define critérios que os metadados das músicas devem cumprir para que elas sejam adicionadas à playlist. Alterações na biblioteca atualizam essas playlists inteligentes automaticamente. Exemplos de Playlists Inteligentes:

- Músicas cuja classificação seja 5 estrelas;
- Músicas de determinado artista, lançadas antes de 1990, que o usuário não tenha ouvido no último ano;
- Músicas ainda sem classificação (Zero estrelas).

1.3.3 Playlists Geradas Automaticamente

A plataforma Spotunes também deve ser capaz de recomendar playlists geradas automaticamente ao usuário. Os metadados de sua biblioteca, bem como outras playlists criadas por ele, servirão de base para essa funcionalidade. Essa questão já foi explorada por Bonnin [3] e possui suporte na API pública do serviço de streaming Spotify [4, 5].

1.4 APIs Públicas

Grande parte dos serviços de streaming fornece uma API publicamente, para que desenvolvedores possam construir novas ferramentas com base no que já existe na aplicação oficial. Isto é exatamente o que se pretende com o desenvolvimento do Spotunes, onde as APIs fornecerão os dados sobre as músicas, assim como a reprodução delas. APIs dos principais serviços:

- Spotify Web API [6];
- Apple Music API [7];
- Youtube Data API [8];
- Soudcloud API [9];
- Deezer API [10];
- FreeSound API [11].

Parte do trabalho consistirá em testar essas APIs de forma a checar suas funcionalidades e ver como podem se encaixar na plataforma, além de verificar como se dará a reprodução por streaming dentro do Spotunes.

1.5 Usuários

Também será implementado um sistema de Usuários, de forma que um utilizador ao efetuar login terá acesso apenas aos dados de sua própria biblioteca. Com isso, será possível desenvolver features sociais, onde usuários podem escolher seguir seus amigos, visualizar o que outros estão escutando no momento, além de compartilhar dados da própria biblioteca, mostrando aos amigos deles suas preferências musicais.

Seria interessante que este trabalho continuasse para além do escopo de um Trabalho de Conclusão de Curso, e que o desenvolvimento desta plataforma proseguisse após o encerramento da disciplina, o que pode ser impulsionado por ser uma plataforma *open source*. Para validar isto, conquistar usuários reais também está nos planos e o entendimento de suas necessidades se dará através de questionários divulgados em aplicativos de mensagens e fóruns, visando entender quais são seus os hábitos de consumo de música.

2 Objetivos

Resumindo os objetivos do trabalho, apresentados e contextualizados na seção anterior:

- Modelar a biblioteca do usuário, armazenando os metadados das músicas incluídas (tanto da faixa musical em si quanto dados de reprodução) em um banco de dados;
- Dar suporte a Playlists Inteligentes e Manuais;
- Permitir o uso das APIs dos serviços principais mais consagrados: Spotify, Youtube e SoundCloud;
- Fornecer um sistema de usuários que possam seguir outros usuários, bem como uma seção de “o que está sendo ouvido no momento ou o que foi ouvido recentemente”;
- Desenvolver um sistema de recomendação de músicas e geração de playlists automáticas com base nas bibliotecas dos usuários;
- Criar um sistema de recomendação de acordo com a biblioteca de amigos próximos.

Alguns objetivos secundários (de caráter de expansão do escopo do sistema), que dependerão do desenvolvimento das features principais acima:

- Caso o desenvolvimento em paralelo das outras features esteja adiantado e bem avançado, implementar suporte às APIs de serviços secundários (os questionários auxiliarão a entender o mais buscados), encapsulando o uso das APIs em uma classe, de forma que usuários possam contribuir no Github com serviços de sua preferência ainda não contemplados;
- Assim como o iTunes é um agregador de mídias além de faixas musicais, a biblioteca do Spotunes pode ser expandida para contar com vídeos e podcasts também.

3 Cronograma Aproximado

Segue abaixo um cronograma aproximado mês a mês das atividades:

Atividade	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov	Dez	Jan
1.	x	x								
2.		x	x							
3.			x							
4.				x						
5.					x	x				
6.						x	x			
7.								x	x	
8.						x	x	x	x	
9.									x	x

Legenda das atividades:

1. Estudos preparatórios e trabalhos preliminares (resumo, proposta e preparação da homepage)
2. Testes com as APIs públicas de vários serviços
3. Implementação da base da biblioteca de usuário com diversos metadados
4. Playlists Inteligentes e Manuais
5. Sistema de Usuários
6. Sistema de Recomendação
7. Refatorações, objetivos secundários e serviços adicionais
8. Preparo da Monografia
9. Preparo da Apresentação

Referências

- [1] Apple, “Gerenciador de biblioteca virtual itunes,” 2021, last accessed 22 May 2021. [Online]. Available: <https://www.apple.com/br/itunes/>
- [2] —, “Create, delete and use smart playlists in itunes on pc,” 2021, last accessed 22 May 2021. [Online]. Available: <https://support.apple.com/en-gb/guide/itunes/itns3001/windows>
- [3] G. Bonnin and D. Jannach, “Automated generation of music playlists: Survey and experiments,” *ACM Computing Surveys*, vol. 47, pp. 1–35, 01 2014.
- [4] J. Lovera, “Magic playlist,” 2021, last accessed 22 May 2021. [Online]. Available: <https://developer.spotify.com/community/showcase/magic-playlist/>
- [5] rob_med, “Build your own playlist generator with spotify’s api,” 2019, last accessed 22 May 2021. [Online]. Available: <https://medium.com/analytics-vidhya/build-your-own-playlist-generator-with-spotifys-api-in-python-ceb883938ce4>
- [6] Spotify, “Spotify web api,” 2021, last accessed 22 May 2021. [Online]. Available: <https://developer.spotify.com/documentation/web-api/>
- [7] Apple, “Apple music api,” 2021, last accessed 22 May 2021. [Online]. Available: <https://developer.apple.com/documentation/applemusicapi/>
- [8] Google, “Youtube data api,” 2021, last accessed 22 May 2021. [Online]. Available: <https://developers.google.com/youtube/v3>
- [9] SoundCloud, “Soundcloud for developers,” 2021, last accessed 22 May 2021. [Online]. Available: <https://developers.soundcloud.com/>
- [10] Deezer, “Api - deezer for developers,” 2021, last accessed 22 May 2021. [Online]. Available: <https://developers.deezer.com/api>
- [11] FreeSound, “Freesound api documentation,” 2021, last accessed 22 May 2021. [Online]. Available: <https://freesound.org/docs/api/>