Musipath: Um Sistema para Exploração de Relacionamentos de Artistas em Redes de Músicas

Musipath: A System for Exploring Artist Relationships in Music Networks

William Rodrigues da Silva
Graduado em Informática pela Universidade Estadual de Maringá - UEM
Universidade Estadual de Maringá
Av. Colombo, 5790 - Maringá - PR
E-mail: ra99514@uem.br

Marcos Aurélio Domingues
Doutor em Ciência de Computadores pela Universidade do Porto - UP
Universidade Estadual de Maringá
Av. Colombo, 5790 - Maringá - PR
E-mail: madomingues@uem.br

RESUMO
A música é um dos principais elementos culturais de uma sociedade, sendo considerada como um fenômeno social, capaz de manter e ditar tradições em sociedades ao longo do tempo. A música não é apenas uma forma de entretenimento, mas está diretamente ligada com a formação de um indivíduo. Já economicamente, destaca-se o crescimento no uso de plataformas de *streaming*, responsáveis pelo crescimento no setor fonográfico. Estas plataformas se mantêm em contínua evolução, buscando melhorar a apresentação do conteúdo musical para os seus usuários. Neste contexto, este trabalho visa contribuir com o aprimoramento das plataformas de *streaming* por meio da proposta e desenvolvimento de um sistema que apresente possíveis soluções para problemas e limitações ocorrentes em *players* de música, facilitando a interação do usuário com artistas e músicas, e viabilizando a busca de relações entre músicas e artistas de seu interesse. Essas relações são construídas e exibidas para o usuário a partir de uma rede de artistas e músicas criada pelo sistema. O método utilizado para exibir relações de similaridade entre artistas foi baseado na criação de grafos que representam tal similaridade, tal que cada artista aparece ligado a um conjunto de artistas similares. Dessa forma, a rede de artistas e músicas criada pelo sistema pode ser explorada pelo usuário por meio dos grafos de cada artista, como também a partir de um sistema de recomendação de álbuns desenvolvido no sistema. O sistema foi avaliado com usuários reais e os resultados mostraram a utilidade dela.

Palavras-Chave: Recuperação de informações musicais, *Players* de música, Similaridade de artistas de música, Grafos, Redes de artistas e músicas, Sistemas de recomendação.

ABSTRACT
Music is one of the main cultural elements of society, being considered as a social phenomenon, able to maintain and dictate traditions in societies over time. In this way, music is not only a form of entertainment, but is directly linked to the formation of an individual. Economically, the growth in the use of streaming platforms stands out, being responsible for the growth in the phonographic sector. These platforms are constantly evolving, seeking to improve the presentation of musical content to its users. In this context, this work aims to

contribute to the improvement of streaming platforms through the proposal and development of a system that presents possible solutions to problems and limitations occurring in music players, facilitating user interaction with artists and songs, and, moreover, enabling search for relationships between songs and artists of interest. These relationships are built and displayed to the user from a network of artists and songs created by the system. The method used to show similarity relationships between artists was based on the creation of graphs that represent such similarity, where each artist appears linked to a set of similar artists. In this way, the network of artists and songs can be explored by the user through the graphs of each artist, as well as through an album recommendation system implemented in the system. The system was evaluated with real users and the results showed its usefulness.

Keywords: Music information retrieval, Music players, Similarity of music artists, Graphs, Networks of artists and music, Recommendation systems.

1 INTRODUÇÃO

A música é um dos principais elementos culturais de uma sociedade. Gregory (1997) considera a música um fenômeno social, sendo capaz de manter e ditar tradições em sociedades ao longo do tempo. A música não é apenas uma forma de entretenimento, mas está diretamente ligada com a formação de um indivíduo. De acordo com North et al. (2000), por meio da música crianças mudam a forma de pensar, agir e estabelecer relações com amigos, colegas, modificando a sua rede social.

Já economicamente, destaca-se o crescimento no uso de plataformas de *streaming* como Spotify[1], Apple Music[2], Pandora[3], dentre outras plataformas, responsáveis pelo crescimento no setor fonográfico. Segundo relatório publicado em 2019 pela Federação Internacional de Indústria Fonográfica (IFPI[4] - *International Federation of the Phonographic Industry*), o comércio no setor de mídias digitais apresentou um aumento na receita de 21,1%, gerando um lucro de US $11,2 bilhões. Já o mercado brasileiro, impulsionado pelo mercado mundial, também apresentou um crescimento de 15,4%, sendo então considerado o maior mercado na América do Sul e o 10º país em escala mundial no consumo de mídias digitais.

Mesmo com um crescimento no uso de plataformas de *streaming*, estas se mantêm em contínua evolução, buscando melhorar a apresentação do conteúdo musical para os seus usuários. Nesse contexto, os *players* de música podem ser vistos como um importante mecanismo de interação e exibição de artistas e músicas com usuários em uma plataforma de *streaming*. Esse tipo de mecanismo apresenta aos usuários listagens de artistas e músicas que

[1] https://www.spotify.com
[2] https://www.apple.com/br/music
[3] https://www.pandora.com
[4] http://www.ifpi.org

possibilitam a interação com estes. Contudo, nem sempre o conteúdo é apresentado ao usuário de forma intuitiva, o que prejudica também o interesse do usuário no uso da plataforma. Como exemplo, em um álbum de músicas, a identificação de artistas e músicas similares ao interesse do usuário pode não ser apresentada, ou não ser informado ao usuário de maneira clara os motivos para a existência de similaridade entre músicas e artistas. É importante mencionar também que as plataformas de *streaming* em sua maioria são ofertadas estabelecendo uma taxa de uso, gerando assim um fator limitante para o acesso a elas.

Tendo em vista o importante papel cultural da música em uma sociedade e, como já mencionado, a função de *players* de músicas como ferramentas para interação de usuários com músicas e seus artistas, este trabalho tem como objetivo principal a proposta e o desenvolvimento de uma ferramenta para visualização e interação com redes de artistas e músicas, adicionando novas funcionalidades a um *player* de música. A representação da rede de artistas e músicas por meio de grafos é inspirada no conceito apresentado por Costa et al. (2009).

O restante deste artigo está organizado da seguinte forma. Na Seção 2 são apresentados alguns exemplos de trabalhos que já existem na literatura, relacionados com o tema deste artigo, mencionando as principais diferenças e vantagens encontradas quando relacionados com este trabalho. Na Seção 3 é apresentada a ferramenta proposta e suas funcionalidades. Na Seção 4, são apresentados os resultados obtidos em uma avaliação realizada com usuários reais. Por fim, na Seção 5 são apresentadas as conclusões e trabalhos futuros.

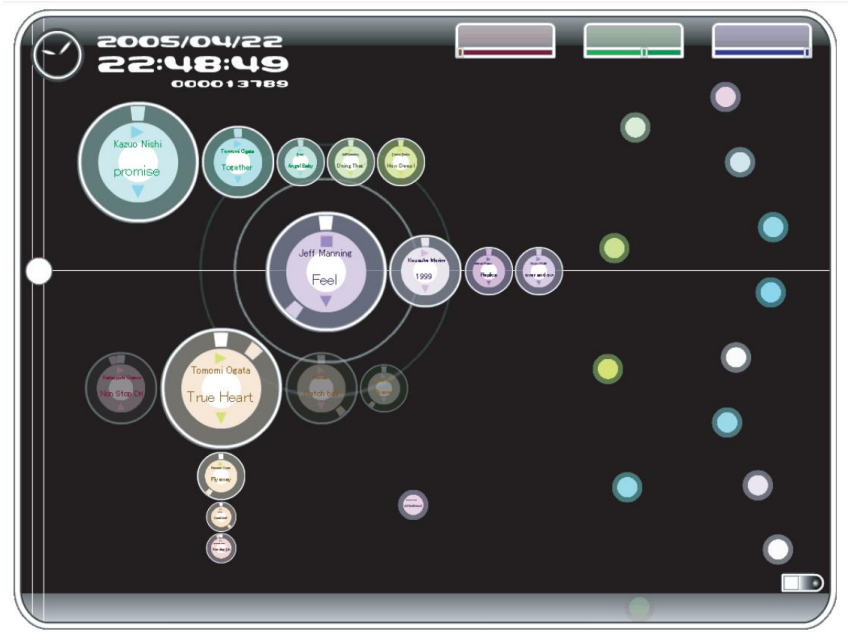
2 TRABALHOS RELACIONADOS

Com o desenvolvimento da área de recuperação de informações musicais, coleções de dados sobre música tornaram-se cada vez maiores com o passar dos anos. Neste contexto, verificou-se a necessidade do desenvolvimento de ferramentas que fossem capazes de selecionar e exibir conjuntos de dados musicais para o usuário de maneira coerente. Nesta seção são apresentados alguns exemplos de trabalhos relacionados e discutidas as diferenças observadas com relação à proposta do presente trabalho.

O *Musicream* é um sistema proposto por Goto, M. e Goto, T. (2009), que tem como fundamento a proposta de uma interface de exibição e reprodução de músicas incomuns, pois tenta desfazer o estereótipo de interfaces que exibem listas compostas por títulos de músicas e seus artistas. A ideia é proporcionar ao usuário uma experiência de descoberta de álbuns de música de forma inesperada, por meio de *playlists* criadas de acordo com o gosto do usuário.

Conforme exposto na Figura 1, os discos na interface representam os álbuns de música, que são reproduzidos de forma sequencial. Através de uma busca inicial por um álbum, o sistema oferece outras opções de álbuns similares que o usuário pode anexar à reprodução.

Figura 1 – Interface de exploração de discos (álbuns) no *Musicream*



Fonte: Goto, M.; Goto, T. Musicream, 2009

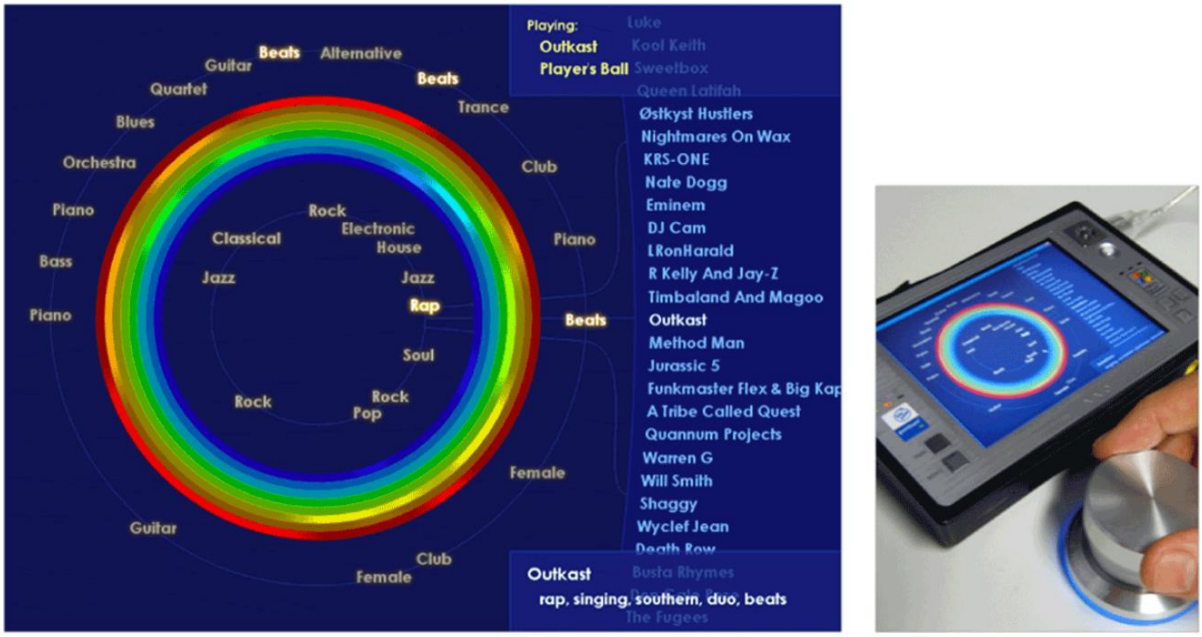
Outro exemplo é o sistema *FM4 Soundpark Player* proposto por Gasser, M. e Flexer, A. (2009), que utiliza como chave inicial de busca uma música específica inserida pelo usuário. Partindo de uma música inicial, o sistema cria um grafo, como exposto na Figura 2, que representa sugestões de músicas similares, sugerindo até cinco músicas que coincidem com o conteúdo da música buscada. O usuário tem então a possibilidade de interagir com o grafo criado, e sempre que um dos vértices do grafo é selecionado, representando determinada música, o grafo é atualizado, determinando o vértice selecionado como central e realizando novas sugestões, partindo deste vértice central. Desta maneira o usuário consegue interagir com sugestões musicais e ouvir cada música através de um *player* de música acoplado no sistema.

Figura 2 – Grafo de músicas e *player* do sistema *FM4 Soundpark*

Fonte: Gasser, M.; Flexer, A. FM4 Soundpark, 2009

Outra abordagem de interface é apresentada no sistema *MusicRainbow* proposto por Pampalk, E. e Goto, M. (2006). O sistema tem como objetivo proporcionar uma experiência de descoberta de novos artistas para o usuário. A disposição dos artistas é mapeada em uma interface que segue a ideia de um arco-íris em formato circular, em que cada cor do arco-íris representa um gosto musical diferente. O sistema conta com um pré-processamento de dados que separa os artistas de acordo com seus rótulos, extraídos de páginas web. O sistema pode resumir e dividir artistas em subgrupos de acordo com os rótulos mais representativos, como é exposto na Figura 3. A interação com o usuário é feita de uma forma simplificada. O usuário pode rotacionar o arco-íris e selecionar os rótulos de sua preferência, obtendo uma lista de artistas correspondentes.

Figura 3 – Interface do sistema *MusicRainbow*



Fonte: Pampalk, E.; Goto, M. MusicRainbow, 2006

Em síntese, podem-se descrever os sistemas propostos nesses trabalhos, a partir dos métodos utilizados na geração de conteúdo para o usuário. No primeiro sistema citado (*Musicream*), a proposta de busca de informações musicais é dada a partir de um álbum inicial informado pelo usuário, e o sistema gera um fluxo de conteúdo sugerindo novos álbuns similares. Já no segundo sistema (*FM4 Soundpark*), a busca inicial é realizada a partir de uma música informada pelo usuário, e o conteúdo gerado é baseado na sugestão de músicas similares. Por fim, no terceiro sistema (*MusicRainbow*), a recomendação de artistas é realizada a partir da comparação dos rótulos de cada um.

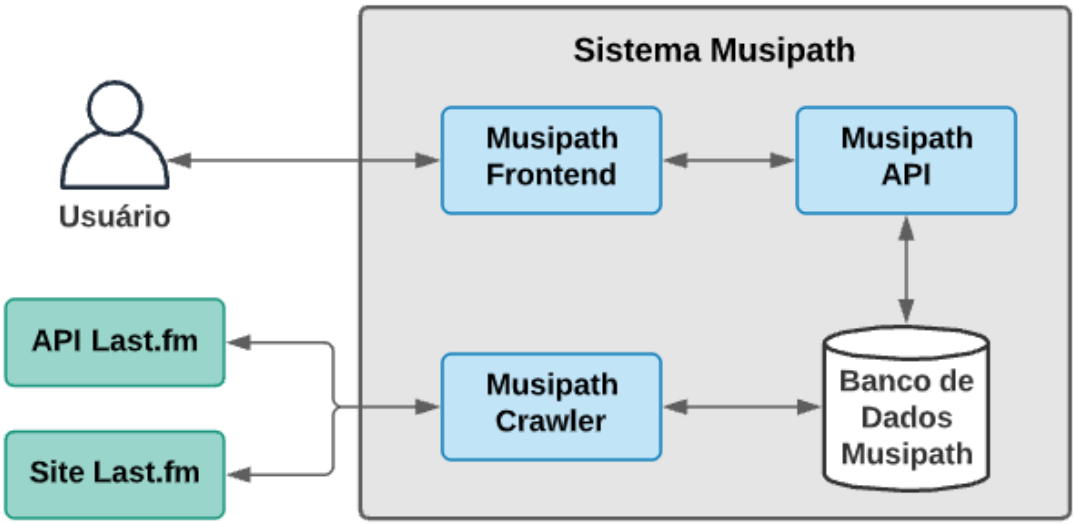
A ferramenta proposta neste trabalho tem um objetivo análogo de permitir ao usuário uma experiência de interação e descoberta de novos artistas, álbuns e músicas. Entretanto, a interface proposta não restringe a exploração do conteúdo apenas a álbuns ou apenas a artistas, mas propõe uma exploração que pode ser feita por ambos os caminhos. O usuário pode interagir e descobrir novos artistas a partir de um grafo interativo que representa a conexão entre artistas similares, mas também consegue visualizar os álbuns de cada artista, recebendo recomendações de álbuns similares que pertencem a outros artistas, com mais possibilidades na exploração da rede de artistas e músicas. O sistema proposto também propõe uma interface simplificada, com objetivo de facilitar a experiência de interação.

3 PROPOSTA E DESENVOLVIMENTO DO SISTEMA *MUSIPATH*

O sistema proposto e desenvolvido neste trabalho foi intitulado como "*Musipath*". O conceito do nome está na ideia da união das palavras em inglês "*music*" e "*path*", que significam respectivamente, "música" e "caminho/trajeto", fazendo uma analogia aos possíveis cursos que o usuário pode tomar em sua experiência musical no uso do sistema.

Pode-se classificar a divisão das funcionalidades propostas no sistema *Musipath* a partir do desenvolvimento de três subsistemas que o compõem: *Musipath Crawler*, *Musipath API* e *Musipath Frontend*. A Figura 4 ilustra uma visão geral sobre o sistema *Musipath*, seus subsistemas, fontes de informação, e a comunicação existente entre esses elementos.

Figura 4 – Visão geral do sistema *Musipath*



Fonte: Elaborada pelo autor, 2021

Nesta seção será apresentado o desenvolvimento dos três subsistemas, explicando como cada funcionalidade proposta foi desenvolvida.

3.1. Pré-processamento dos Dados

O pré-processamento de dados compreendeu a etapa inicial de desenvolvimento do sistema. Nesta seção será apresentada a implementação do subsistema *Musipath Crawler*, que tem como principal objetivo a recuperação de informações musicais para construção da base de dados do sistema *Musipath*. O banco de dados adotado para o sistema foi o *MySQL*[5] e a manipulação deste é realizada pela ferramenta de mapeamento *Hibernate*[6].

[5] https://www.mysql.com
[6] https://hibernate.org

A necessidade de recuperação de informações musicais neste contexto pode ser descrita especificamente como a obtenção de dados sobre artistas de música, álbuns, músicas e existência de similaridade entre artistas.

O subsistema foi desenvolvido sob a linguagem de programação *Java*[7], aplicando o conceito de orientação a objetos, e seus métodos para recuperação de informações musicais foram separados em duas classes principais, de acordo com a fonte da qual as informações são extraídas, nomeadas como *CrawlerAPI* e *CrawlerWeb*.

Para explicar o funcionamento da classe *CrawlerAPI* é preciso entender primeiramente o conceito de *Application Programming Interface* (API), que pode ser definido como um conjunto de rotinas implementadas por determinado sistema, com o objetivo de oferecer funcionalidades a serem utilizadas por outros sistemas. Desta forma uma API permite a comunicação e integração entre sistemas, sem haver a necessidade de que um sistema conheça a implementação do outro.

A classe *CrawlerAPI* tem como fonte de extração de dados a API disponibilizada pelo site Last.fm[8], uma plataforma musical colaborativa, que permite que seus usuários classifiquem artistas, álbuns e músicas através de rótulos, utilizando essas informações para definir dimensões semânticas no contexto musical. A API disponibilizada pelo site Last.fm oferece métodos de recuperação de informação musical. Um exemplo de informação recuperada por meio desta API são os relacionamentos de similaridade entre artistas. Uma vez realizada a requisição para obter informações de um determinado artista, tem-se como resposta uma lista de artistas similares a este, uma informação de grande importância na construção da base de dados do sistema *Musipath*. A classe *CrawlerAPI* pode realizar requisições à API do site Last.fm, que retorna respostas no formato JSON[9], contendo informações úteis para construção dos objetos que representam artistas, álbuns e músicas no sistema.

Entretanto, nem todas as informações necessárias para criação destes objetos puderam ser obtidas utilizando a API do site Last.fm, como a URL de uma música no site do Youtube[10], a imagem de capa de um álbum, dentre outras. Assim, a criação de métodos para obtenção dessas informações se tornou necessária.

A classe *CrawlerWeb* encapsula os métodos implementados com a finalidade de obter o restante das informações necessárias, porém ausentes na recuperação de informações da

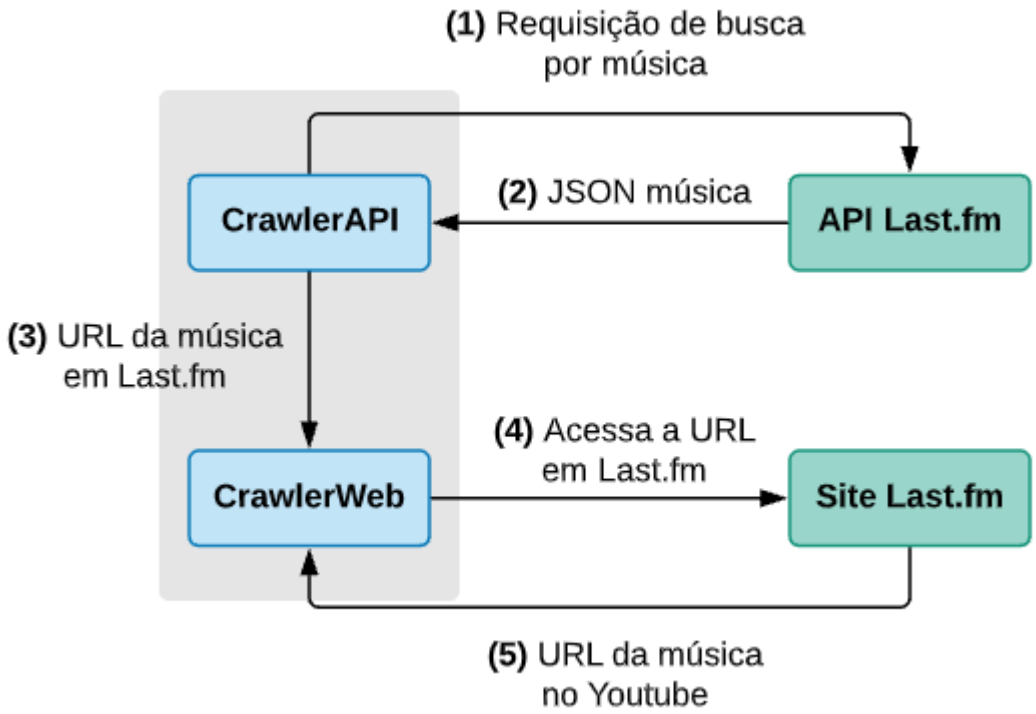
[7] https://www.java.com
[8] https://www.last.fm
[9] https://www.json.org
[10] https://www.youtube.com

classe *CrawlerAPI*. A fonte de extração de informações dessa classe é o próprio site Last.fm. Os métodos implementados acessam páginas web do site e realizam a busca de dados na estrutura da página para recuperação de informações.

É importante ressaltar que alguns métodos presentes na classe *CrawlerWeb* necessitam de informações extraídas previamente pela classe *CrawlerAPI*. Para exemplificar, pode-se citar o contexto de recuperação de informações de uma música.

Como ilustrado na Figura 5, a recuperação de informações de uma música se inicia com uma requisição feita pela classe *CrawlerAPI* à API do site Last.fm. A API retorna uma resposta no formato JSON contendo as informações de uma música, como: Nome da música e URL da música no site Last.fm. Essa informação é então processada e registrada no sistema. Entretanto, o sistema *Musipath* precisa também da URL da música no Youtube, que não é fornecida pela API do Last.fm. Para isso, a classe *CrawlerWeb* utiliza a URL da música no site Last.fm já coletada, e realiza uma busca sobre o conteúdo desta página com a finalidade de obter a URL da música no Youtube. O objeto que representa uma música pode ser criado a partir destas informações extraídas em conjunto.

Figura 5 – Fluxo de recuperação de informações de uma música

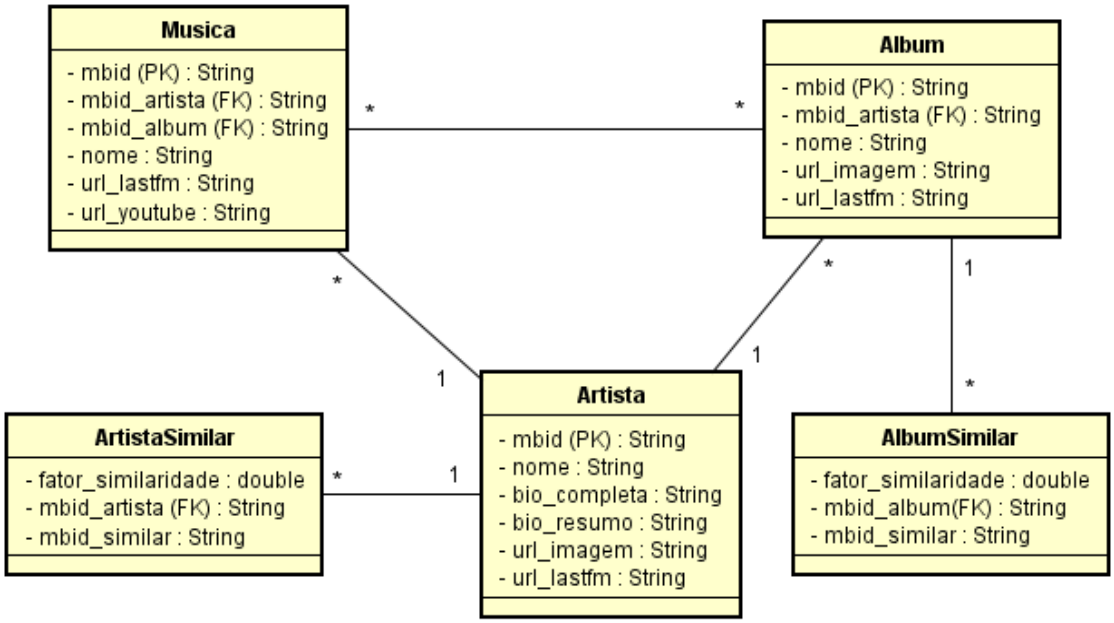


Fonte: Elaborada pelo autor, 2021

Outra funcionalidade dos métodos implementados na classe *CrawlerWeb* é a recuperação de informações como a lista de álbuns de determinado artista, a lista de músicas de determinado álbum e a lista de artistas representados por uma categoria, tendo em vista que essas informações são necessárias para a construção da base de dados do sistema *Musipath*, mas não são informados pela API do site Last.fm.

Com a união dos métodos implementados nas duas classes citadas é possível reunir as informações recuperadas e construir os objetos e suas relações propostas para a base de dados do sistema, como ilustra a modelagem na Figura 6.

Figura 6 – Diagrama de classes do sistema *Musipath*



Fonte: Elaborada pelo autor, 2021

3.2. API de Comunicação do Sistema *Musipath*

Com o desenvolvimento do primeiro subsistema (*Musipath Crawler*) e suas funcionalidades descritas na subseção anterior, o banco de dados do sistema *Musipath* pode ser preenchido com informações sobre artistas, álbuns e músicas. Através dos métodos de processamento dessas informações também é possível estabelecer relações de similaridade entre artistas e álbuns de música.

O segundo subsistema desenvolvido foi nomeado como *Musipath API*. Desenvolvido também sobre a linguagem de programação *Java* com auxílio do framework *Spring Boot*[11]. O

[11] https://spring.io/projects/spring-boot

principal objetivo desse subsistema foi estabelecer um meio de comunicação entre as informações coletadas e registradas no banco de dados, e o subsistema Web de interface com o usuário. O acesso a essas rotinas pode ser feito via *endpoints*, ou pontos de extremidade, os terminais de conexão entre o serviço disponibilizado pela API e o sistema que está consumindo esse serviço. Para exemplificar, pode-se citar o *endpoint* de acesso às informações de um artista, que pode ser feito por meio de uma requisição que segue o modelo: http://{endereço da aplicação}/api/artistas/buscarpornome/{nome}. Neste caso, o subsistema de interface do usuário pode fazer uma requisição à API, obtendo como resposta as informações que compõem um artista, no formato JSON.

3.3. Interface com o Usuário

O terceiro subsistema foi o *Musipath Frontend*, um sistema Web desenvolvido com o uso da ferramenta *React*[12], uma biblioteca *Javascript* declarativa, eficiente na criação de interfaces de usuário. Esta biblioteca permite a criação de interfaces através da criação de componentes. A finalidade deste terceiro subsistema é disponibilizar as funcionalidades propostas inicialmente.

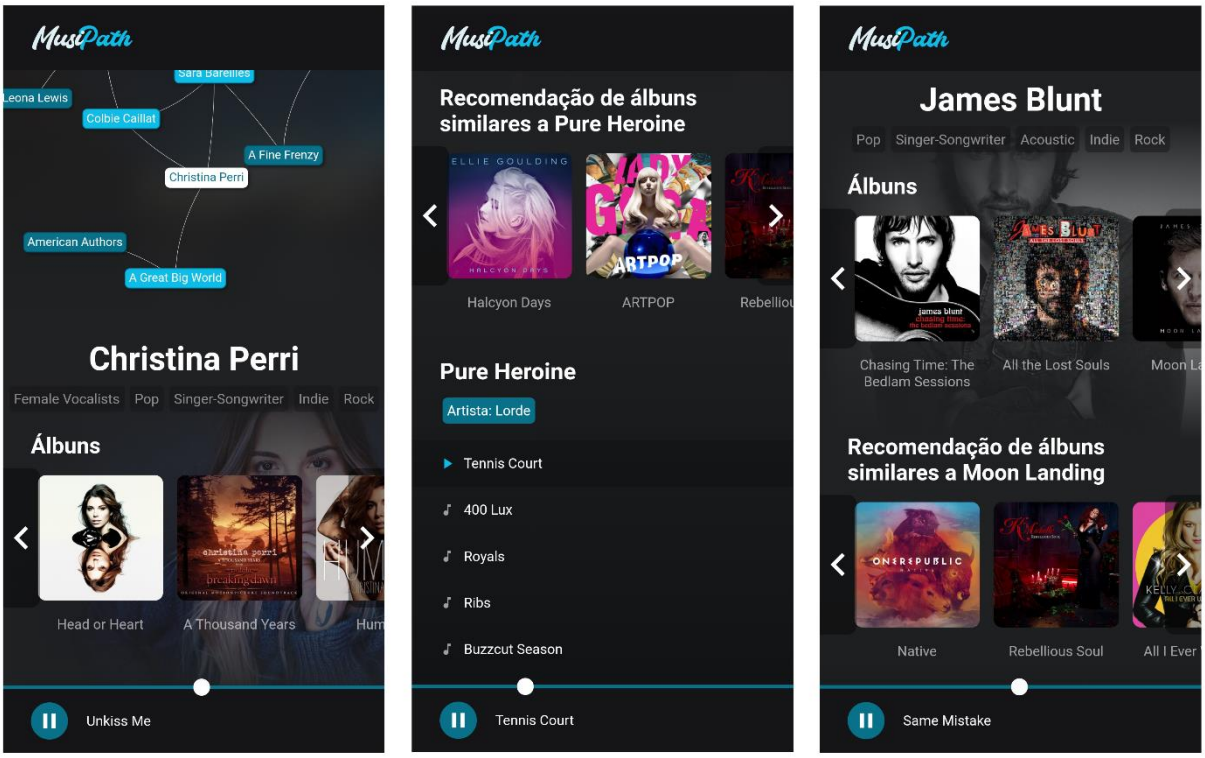
O subsistema Web foi desenvolvido com a proposta de responsividade, isto é, o sistema tem a capacidade de se adaptar e exibir as informações para o usuário em diferentes tipos de dispositivos e resoluções de tela. A Figura 7 exibe exemplos de como a interface é modelada em um dispositivo móvel.

A construção da interface baseada em componentes é uma funcionalidade oferecida pela ferramenta *React*, que permite ao usuário uma interação mais fluida com o sistema, pois os componentes podem ser instanciados e atualizados dinamicamente, sem a necessidade do carregamento de novas páginas. Foram implementados diferentes componentes para compor a interface, cada um responsável por exibir um conjunto de informações, por exemplo, o grafo de similaridade de um artista, informações sobre um artista, informações sobre um álbum, o *player* de música, recomendação de álbuns, dentre outros.

Neste subsistema foi utilizada também uma API do Youtube, com o objetivo de recuperar o áudio das músicas quando selecionadas pelo usuário na interface. Assim, o sistema pode obter o áudio completo de músicas em tempo de execução, durante a interação do usuário.

[12] https://reactjs.org

Figura 7 – Exemplos de interfaces do usuário em um dispositivo móvel



Fonte: Sistema *Musipath*, 2021

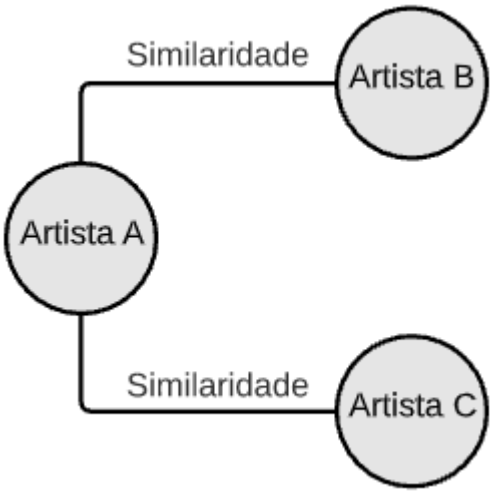
3.4. Criação do Grafo de Artistas

Um dos objetivos do sistema *Musipath* é oferecer ao usuário a experiência de exploração de uma rede de artistas e músicas. Essa exploração pode levar à descoberta de novos artistas, álbuns e músicas que correspondem ao gosto musical do usuário. Uma das ferramentas utilizadas para gerar essa interação foi à criação de uma rede de artistas baseada na similaridade existente entre eles.

Para representar a rede de artistas foram utilizados grafos (BOLLOBÁS, B, 1998). A estrutura de um grafo permite a representação de relacionamentos, no caso, a similaridade entre os artistas.

Como ilustrado na Figura 8, cada artista pode ser representado na rede como um vértice. As arestas, por sua vez, representam a existência de similaridade entre artistas. De acordo com o exemplo de grafo apresentado pode-se concluir que o artista A é similar aos artistas B e C.

Figura 8 – Grafo representando a similaridade entre artistas



Fonte: Elaborada pelo autor, 2021

Para poder exibir a rede de artistas na interface do usuário, foi adotada a estratégia de geração do grafo individual para cada artista. Na prática isso significa que o grafo exibido para o usuário é gerado a partir de um artista inicial, e recursivamente os artistas similares vão sendo adicionados ao grafo, até que o limite de profundidade de exploração definido seja atingido. A profundidade representa o nível de recursão pelo qual o grafo foi gerado, assim, quanto maior a profundidade, maior será o número de vértices adicionados ao grafo. Desta maneira, para cada artista buscado ou selecionado pelo usuário na interface, um grafo é gerado e apresentado, partindo do artista inicial selecionado.

A Figura 9 exibe um pseudocódigo que representa a ideia da função recursiva de geração de grafo. A função recebe como argumentos um artista inicial e a profundidade limite, e devolve como resposta o grafo de similaridade gerado para o artista inicial.

É apresentado como exemplo na Figura 10, um grafo gerado pelo sistema, tendo como artista inicial "*Coldplay*". As cores adotadas na representação gráfica do grafo diferenciam o nível de similaridade entre os artistas. O artista inicial é representado pela cor branca, os artistas que estão diretamente associados a ele, representando uma relação direta de similaridade, são representados por um tom de azul mais claro, e os demais artistas, representados por um tom de azul mais escuro.

Figura 9 – Pseudocódigo para a função de criação do Grafo de um artista

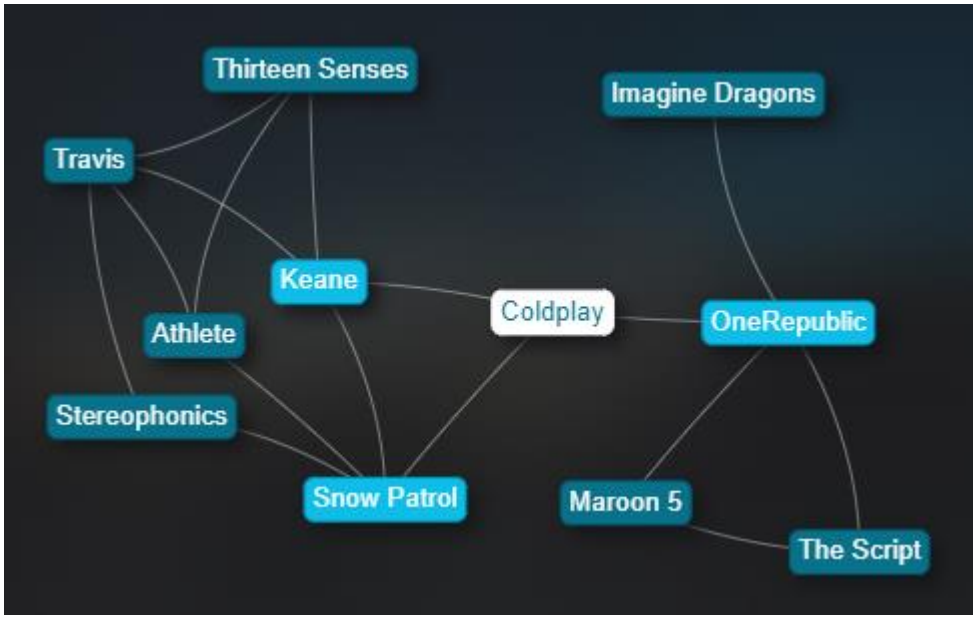
Algoritmo 1 *Algoritmo recursivo para gerar o Grafo de similaridade de um Artista*

Entrada *Grafo, Artista, Profundidade Atual, Profundidade Limite*
Saída *Grafo*

```
Função Gerar Grafo
    se Profundidade Atual ≥ Profundidade Limite então:
        Novo nó ← Criar Nó (Artista)
        Grafo ← Adicionar nó (Grafo, Novo nó)
        retornar Grafo
    fim se
    Profundidade Atual ← Profundidade Atual + 1
    Novo nó ← Criar Nó (Artista)
    Grafo ← Adicionar nó (Grafo, Novo nó)
    para cada Artista Similar a Artista faça:
        Nova aresta ← Criar Aresta (Artista, Artista Similar)
        Grafo ← Adicionar aresta (Grafo, Nova aresta)
        Grafo ← Gerar Grafo (Grafo, Artista Similar, Profundidade Atual, Profundidade Limite)
    fim para
    retornar Grafo
```

Fonte: Elaborada pelo autor, 2021

Figura 10 – Grafo de um artista gerado pelo sistema *Musipath*



Fonte: Sistema *Musipath*, 2021

3.5. Sistema de Recomendação

Por meio da criação do grafo de similaridade de artistas, o sistema possibilita ao usuário a interação e descoberta de artistas similares ao seu gosto musical. Por outro lado, o sistema também propõe um mecanismo de recomendação de álbuns. Assim, o usuário pode

encontrar outros caminhos na descoberta de um conteúdo musical que corresponde ao seu gosto. Por consequência, a descoberta de novos álbuns de música e o contato com novos artistas, os autores de cada álbum.

Conforme Sundermann (2015), sistemas de recomendação são ferramentas utilizadas com o intuito de encontrar e fornecer ao usuário itens de seu interesse, dentro de um conjunto de possíveis itens. Ricci, Rokach e Shapira (2011) ainda definem sistemas de recomendação como uma tecnologia empregada no filtro de informações, que podem ser utilizadas para recomendação de itens através da criação de um *ranking*, e ainda prever de acordo com os interesses do usuário, avaliações de itens.

Sistemas de recomendação tipicamente expressam sua funcionalidade de recomendação fundamentada na relação existente entre usuário e item, onde o usuário corresponde ao alvo de uma recomendação e o item compreende as opções que podem ser sugeridas para este, sendo assim definida a recomendação como uma lista ordenada de itens que correspondem ao interesse do usuário (ADOMAVICIUS; TUZHILIN, 2005; RICCI; ROKACH; SHAPIRA, 2011; BOBADILLA et al., 2013).

O sistema de recomendação adotado para realizar as recomendações de álbuns é baseado no conteúdo de cada álbum. Cada álbum possui um conjunto de rótulos associado a ele. Esse conjunto de rótulos tem como objetivo descrever o conteúdo do álbum através de palavras chaves ou pequenas frases. Os rótulos podem referenciar o gênero de um álbum, instrumentos musicais utilizados, uma época, influências musicais, dentre outros aspectos. Esse conjunto de rótulos que descreve cada álbum foi utilizado para encontrar a similaridade entre álbuns, e assim, realizar recomendações.

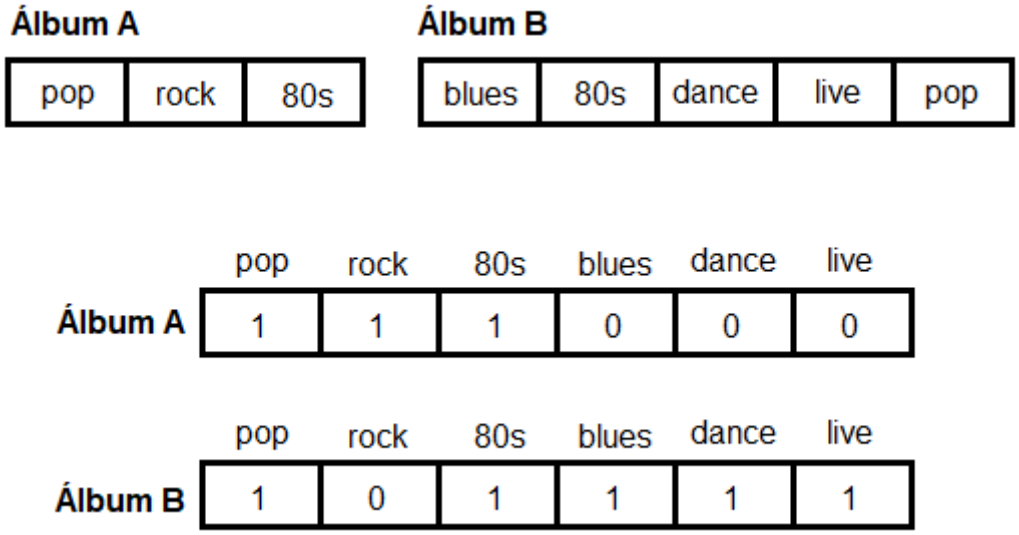
Para determinar o fator de similaridade entre dois álbuns foi adotada a fórmula de similaridade de cosseno definida a seguir:

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}} \tag{1}$$

Para a aplicação da fórmula (1) de similaridade de cosseno, é necessária uma pré-organização de dados. Um exemplo desta organização é apresentado na Figura 11. Inicialmente, cada álbum é representado por um vetor em que cada posição é preenchida com um rótulo do álbum. Para cada um dos álbuns é criado um novo vetor de tamanho N, sendo N

a soma dos rótulos presentes nos vetores iniciais dos dois álbuns, excluindo repetições. Desta forma têm-se dois vetores de tamanhos iguais. Cada posição deste novo vetor representa um rótulo, e será preenchido com 1 caso o rótulo correspondente pertença ao conjunto de rótulos do álbum, e 0 caso contrário. O cálculo de similaridade pode então ser aplicado sobre esses dois novos vetores, obtendo-se um fator normalizado de similaridade entre os dois álbuns. Assim, conclui-se que os dois álbuns são similares quanto mais este fator aproxima-se de 1.

Figura 11 – Organização de vetores para aplicação da similaridade de cosseno



Fonte: Elaborada pelo autor, 2021

Com a implementação do algoritmo de similaridade de cosseno, pode-se obter o fator de similaridade existente entre dois álbuns. Para concluir o sistema de recomendação de álbuns, foi realizado um processo de aplicação deste algoritmo em todos os álbuns do banco de dados do sistema, de forma que cada álbum pudesse ser comparado com todos os demais. Assim, definiu-se para cada álbum, um conjunto de álbuns que tem o maior fator de similaridade com este, sendo esse o conjunto de álbuns que será recomendado para o usuário.

Na próxima seção são apresentados os resultados de uma avaliação realizada com o sistema *Musipath*.

4 AVALIAÇÃO E RESULTADOS

Para fins de avaliação, o sistema foi disponibilizado online e algumas pessoas puderam utilizá-lo e avaliá-lo por meio de um formulário com perguntas e respostas. O formulário foi composto por 5 questões e as respostas foram padronizadas em 5 níveis:

“Ótima”, “Boa”, “Razoável” e “Péssima”, visando identificar as seguintes dimensões: Compreensão do sistema, facilidade de uso, descoberta de artistas e músicas correspondentes aos gostos do usuário, visibilidade das recomendações e utilidade das recomendações.

A avaliação foi realizada com 15 pessoas que aceitaram colaborar na avaliação, dentre as quais 5 já cursavam o ensino superior na área de Informática. Sendo 40% do sexo masculino e 60% feminino, além disso, 73,3% informaram que pertencem a faixa etária entre 15 e 30 anos, 20% entre 31 e 45 anos, e 6,7% entre 46 e 60 anos. Os resultados da avaliação são apresentados na Tabela 1.

Tabela 1 – Resultados da avaliação do sistema *Musipath*

Questões	**Respostas**					
	Ótima	Boa	Razoável	Ruim	Péssima	**Total**
“Como você classifica a compreensão do funcionamento do sistema como um todo?”	10	5	0	0	0	**15**
“Como você classifica a facilidade de uso do sistema?”	9	6	0	0	0	**15**
“Como você classifica a capacidade do sistema quanto a descoberta de artistas ou músicas que correspondem ao seu gosto?”	3	10	2	0	0	**15**
“Como você classifica a visibilidade das recomendações de álbuns? (Você percebeu que existem recomendações?)”	9	6	0	0	0	**15**
“Como você classifica a utilidade das recomendações de álbuns? (Você achou a recomendação útil?)”	4	11	0	0	0	**15**

Fonte: Elaborada pelo autor, 2021

Considerando as respostas apresentadas pelos usuários nesta avaliação, observa-se que 66,7% classificam a compreensão do funcionamento do sistema como ótima, e 33,3% como

boa. Quanto à facilidade de uso do sistema 60% dos usuários consideram como ótima, e 40% como boa. Além disso, 66,7% dos usuários consideram como boa a capacidade do sistema na descoberta de artistas e músicas que correspondem ao seu gosto, 20% como ótima, e 13,3% como razoável. Quanto à visibilidade das recomendações de álbuns, 60% dos usuários consideram como ótima, e 40% como boa. Por fim, 73,3% dos usuários classificaram a utilidade das recomendações de álbum útil como boa, e 26,7% como ótima.

Observa-se que a terceira pergunta, relacionada à classificação do sistema quanto à capacidade de descoberta de novos artistas e músicas, apresentou a maior porcentagem de avaliação "Razoável". Tal resultado pode ser entendido pelo fato de que a versão do sistema disponibilizado para avaliação ainda não possuía um número expressivo de artistas e álbuns registrados em seu banco de dados, tendo em vista que o sistema *Musipath Crawler* precisa de tempo de processamento para recuperação das informações. Aproximadamente 850 artistas, 3000 álbuns e 27000 músicas foram registradas para essa etapa de avaliação. Essa quantidade pode não ser expressiva o suficiente para representar uma grande diversidade de gêneros musicais existentes.

5 CONCLUSÃO E TRABALHOS FUTUROS

Este trabalho compreendeu a proposta e desenvolvimento de sistema, chamado *Musipath*, que realiza a recuperação de informações musicais e permite a interação e exploração de uma rede de artistas e músicas. O sistema permite que usuários explorem a similaridade existente entre artistas de músicas, descobrindo novos artistas, álbuns e músicas de seu interesse. Além disso, o sistema desenvolvido propõe uma interface de fácil entendimento e utilização por parte do usuário.

A elaboração do sistema como um todo foi dividida na implementação de três subsistemas responsáveis por diferentes tarefas, como a recuperação de informações musicais, a criação de uma rede de artistas e músicas, e a criação da interface com o usuário.

Na avaliação do sistema a quantidade de usuários avaliados ainda é pequena, mas indica que o sistema apresenta suas funcionalidades de forma descomplicada para os usuários avaliados, sendo útil na exploração da rede musical criada, e também, efetiva na exploração de novos artistas e músicas que correspondem aos interesses do usuário.

Para trabalhos futuros propõem-se a implementação de novas funcionalidades no sistema, como a criação de *playlists* de música do usuário; e a possibilidade de favoritar músicas, álbuns e artistas do interesse do usuário, criando-se também uma funcionalidade para visualização destas *playlists*, músicas, álbuns e artistas favoritados.

REFERÊNCIAS

ADOMAVICIUS, G.; TUZHILIN, A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. **IEEE transactions on knowledge and data engineering**, IEEE, v. 17, n. 6, p. 734–749, 2005.

BOLLOBÁS, B. **Modern Graph Theory**, Springer-Verlag New York, 1998. p. 1-2.

BOBADILLA, J. et al. Recommender systems survey. **Knowledge-based systems**, Elsevier, v. 46, p. 109–132, 2013.

COSTA, D.; SARMENTO, L.; GOUYON F. RAMA: **An Interactive Artist Network Visualization Tool**. Late-break/demo at ISMIR 2009.

GASSER, M.; FLEXER, A. **FM4 Soundpark: Audio-based music recommendation in everyday use**. In Proceedings of the 6th Sound and Music Computing Conference (SMC), 2009.

GOTO, M.; GOTO, T. **Musicream: Integrated music-listening interface for active, flexible, and unexpected encounters with musical pieces**. *IPSJ (Information Processing Society of Japan)*, 2009.

GREGORY, A. H. **The roles of music in society: The ethnomusicological perspective**. [S.l.]: Oxford University Press, 1997.

NORTH, A. C.; HARGREAVES, D. J.; O'NEILL, S. A. **The importance of music to adolescents. British Journal of Educational Psychology**, Wiley Online Library, v. 70, n. 2, p. 255–272, 2000.

PAMPALK, E.; GOTO, M. **MusicRainbow: A new user interface to discover artists using audiobased similarity and web-based labeling**. In Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR), 2006.

RICCI, F.; ROKACH, L.; SHAPIRA, B. Introduction to recommender systems handbook. In: **Recommender systems handbook**. [S.l.]: Springer, 2011. p. 1–35.

SUNDERMANN, C. V. **Extração de informação contextual utilizando mineração de textos para sistemas de recomendação sensíveis ao contexto**. Dissertação (Mestrado) — Instituto de Ciências Matemáticas e de Computação (ICMC) - USP, São Carlos, 2015.