

**USJT – UNIVERSIDADE SÃO JUDAS TADEU**

**PROFESSORA:**

**ANDREIA CRISTINA GRISOLIO MACHION**

**ALUNOS:**

**JOÃO DANIEL SALVADO CASTRO – RA: 819164876**

**LUCAS CAYRES DE SANTANA – RA: 820279348**

**PAMELA ROZOLEM – RA: 820279815**

**RAFAEL FERNANDES MALAFAIA LEITE – RA: 820279781**

**RAMIREZ SOARES LIMA – RA: 820150127**

**THUANE DAVID CUNHA – RA: 818119726**

**MODELAGEM DE SOFTWARE**

**SATURN - SISTEMA DE RECOMENDAÇÃO DE MÚSICAS**



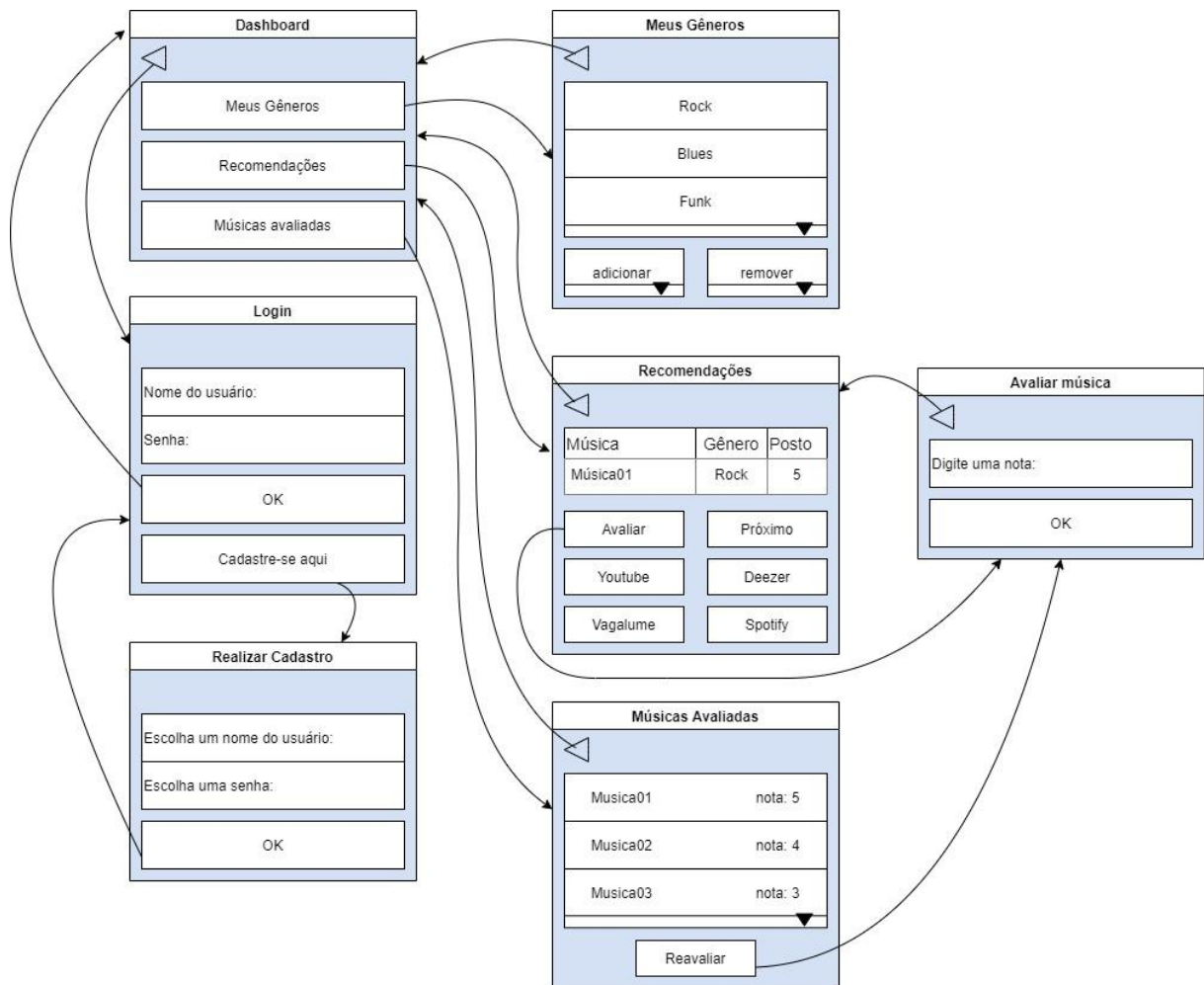
**SÃO PAULO**

**2020**

## INTRODUÇÃO

Este trabalho é a parte da documentação do sistema a ser implementado para o projeto de conclusão parcial das unidades curriculares. É um sistema de recomendação de músicas. Cada usuário do sistema avalia quantas músicas quiser e com base nas avaliações feitas pelos demais usuários, um usuário poderá requisitar uma lista ordenada de músicas recomendadas.

## WIREFRAME



De acordo com o enunciado do Projeto do Sistema de Recomendação de Músicas enviado pelos professores, o sistema que vai ser criado pelos alunos não necessariamente teria que ter as telas do wireframe do enunciado, sendo assim, o sistema criado para este trabalho esta com as telas do wireframe acima que são um pouco diferentes das telas do wireframe do enunciado. Portanto o documento de requisitos e os diagramas apresentados neste trabalho são baseados no enunciado, porém com o wireframe acima.

## DOCUMENTO DE REQUISITOS

### Requisitos Funcionais (RF)

RF01: O sistema deve poder manter(CRUD) todas as músicas do sistema em uma coleção de músicas do sistema

RF01: CRUD são operações que poderão adicionar, consultar, atualizar ou remover as músicas da coleção

RF02: O sistema deve poder manter(CRUD) todos os gêneros do sistema em uma coleção de gêneros do sistema

RF02: CRUD são operações que poderão adicionar, consultar, atualizar ou remover os gêneros da coleção

RF03: O sistema deve permitir o cadastro de usuários

RF03: Assim que o usuário se cadastra, o usuário será armazenado em uma coleção de usuários do sistema

RF04: O sistema deve poder manter(CRUD) todos os usuários do sistema em uma coleção de usuários do sistema

RF04: CRUD são operações que poderão adicionar, consultar, atualizar ou remover os usuários da coleção

RF05: O sistema deve permitir aos usuários fazerem login no sistema

RF05: O sistema deve permitir somente aos usuários que tiverem cadastro fazerem login no sistema, usando nome de usuário e senha

RF06: O sistema deve poder apresentar para o usuário uma lista de músicas que já foram avaliadas pelo próprio usuário, mostrando as notas das suas avaliações

RF07: O sistema deve permitir ao usuário reavaliar as músicas da lista de músicas que já foram avaliadas pelo próprio usuário

RF08: O sistema deve garantir que uma música tenha pelo menos um gênero associado

RF09: O sistema deve permitir que uma música tenha dois gêneros associados

RF10: O sistema deve permitir que o usuário avalie quantas músicas quiser

RF10: O sistema deve permitir que o usuário dê uma nota de 1 a 5 para cada música que ele avaliar, sendo quanto maior a nota, maior o gosto do usuário pela música

RF11: O sistema deve permitir que o usuário requisiite uma lista de músicas recomendadas

RF11: O sistema deve poder apresentar uma lista de músicas recomendadas para cada usuário, com músicas que o usuário ainda não avaliou, em ordem decrescente por posto de música e sendo as músicas associadas á pelo menos um gênero que o usuário tem na sua coleção de gêneros do usuário

RF12: O sistema deve atribuir um posto para cada musica

RF12: O posto de cada música será igual ao valor da média das notas que a música recebeu de cada usuário existente no sistema

RF13: O sistema deve permitir para cada usuário uma coleção de gêneros do usuário

RF13: O sistema deve permitir ao usuário poder adicionar, consultar, remover e atualizar(CRUD) gêneros de sua própria coleção, sem comprometer as coleções de outros usuários

RF14: O sistema deve permitir ao usuário abrir uma música da lista de músicas recomendadas em uma página da internet para ouvir a música

RF14: O sistema deve permitir abrir uma música da lista de músicas recomendadas em uma das plataformas que o usuário escolher, sendo elas: Youtube, Deezer, Spotify e Vagalume

## **Requisitos Não Funcionais (RNF)**

### Requisitos do Produto

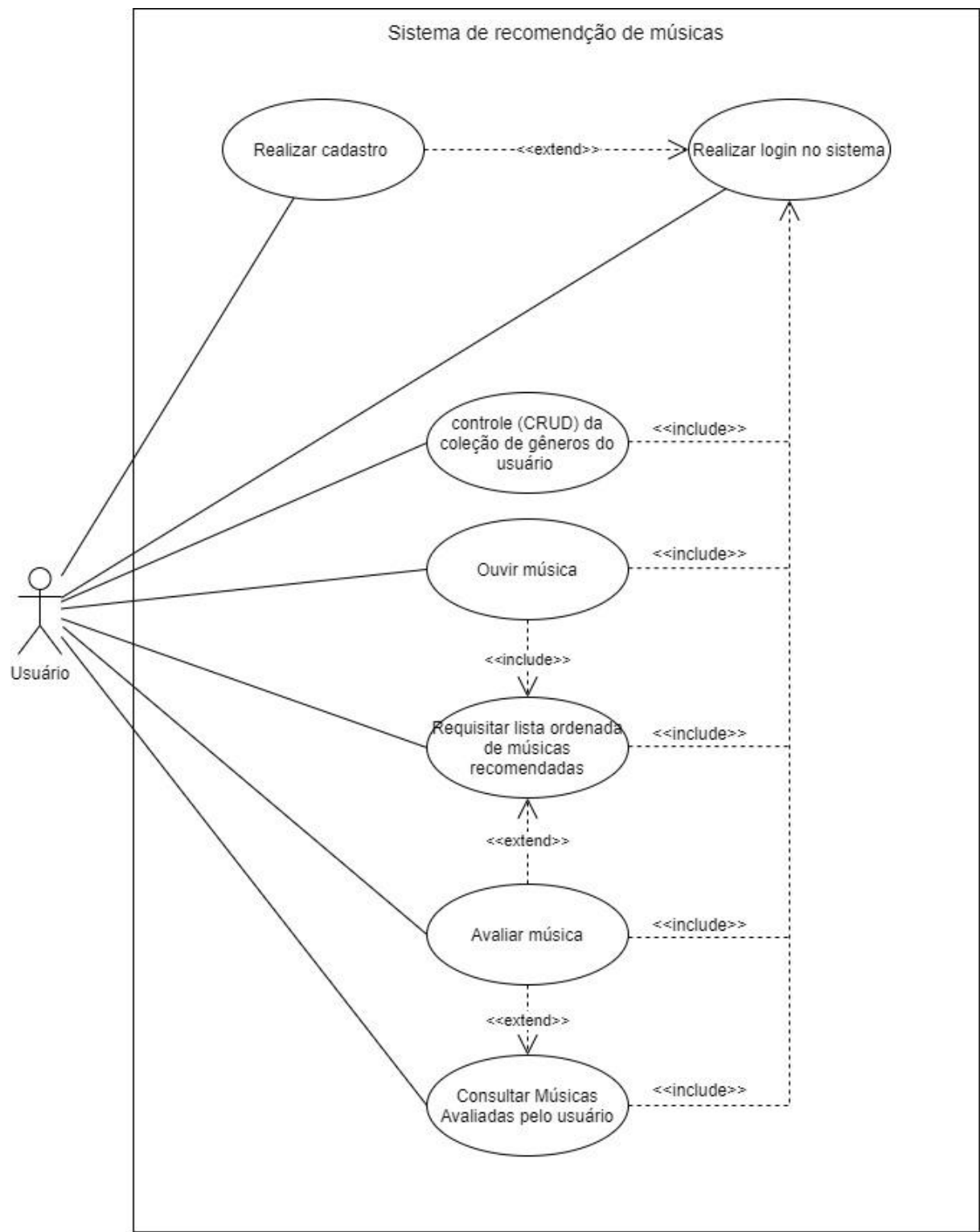
RFN1: Usabilidade: As telas devem facilitar a escolha do destino

### Requisitos Organizacionais

RNF2: Implementação: O sistema deve ser implementado em linguagem Java com o paradigma de orientação a objetos

RNF3: Implementação: O sistema deve armazenar suas informações em um banco de dados com o paradigma relacional de representação de dados em meio persistente usando o MySQL Workbench

DIAGRAMA DE CASOS DE USO



- Quando o usuário for Realizar Login ele tem opção de Realizar cadastro, se não tiver cadastrado
- Para o usuário fazer todas as operações fora Realizar cadastro ele precisa estar logado no sistema (Realizar Login)
- Controle(CRUD) da coleção de gêneros do usuário, permite o usuário operações de adicionar, consultar, atualizar ou remover gêneros da sua coleção de gêneros
- Quando o usuário Requisitar lista de músicas recomendadas, na tela de Recomendações o usuário tem a opção de Avaliar música ou Ouvir música
- O usuário só pode solicitar Ouvir música pela tela de Recomendações
- Quando o usuário Consultar Músicas que ele já avaliou, ele tem opção de Avaliar música novamente nessa tela



DIAGRAMA DE ATIVIDADES

Atividade *core* do sistema: Requisitar lista de músicas recomendadas

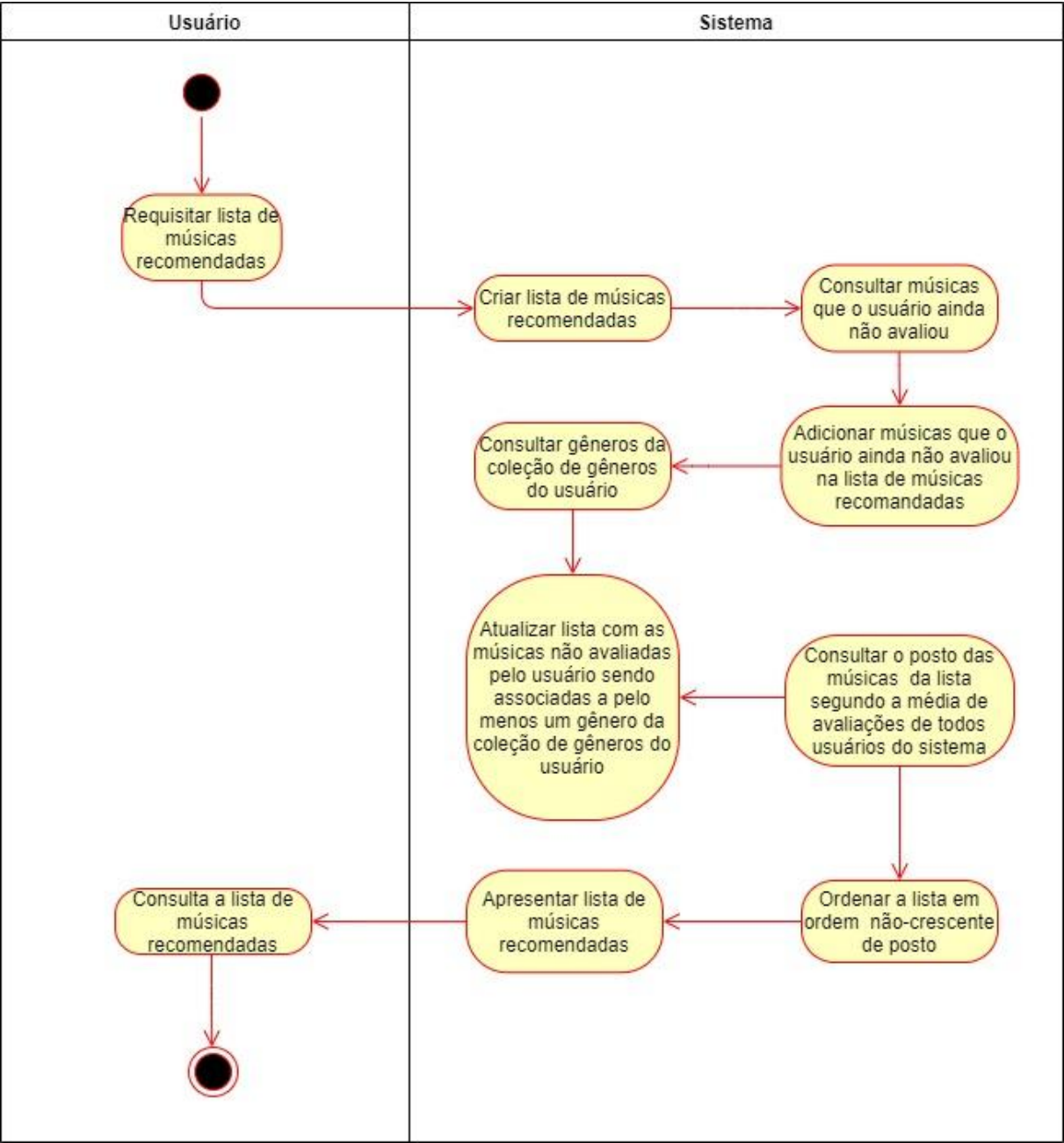
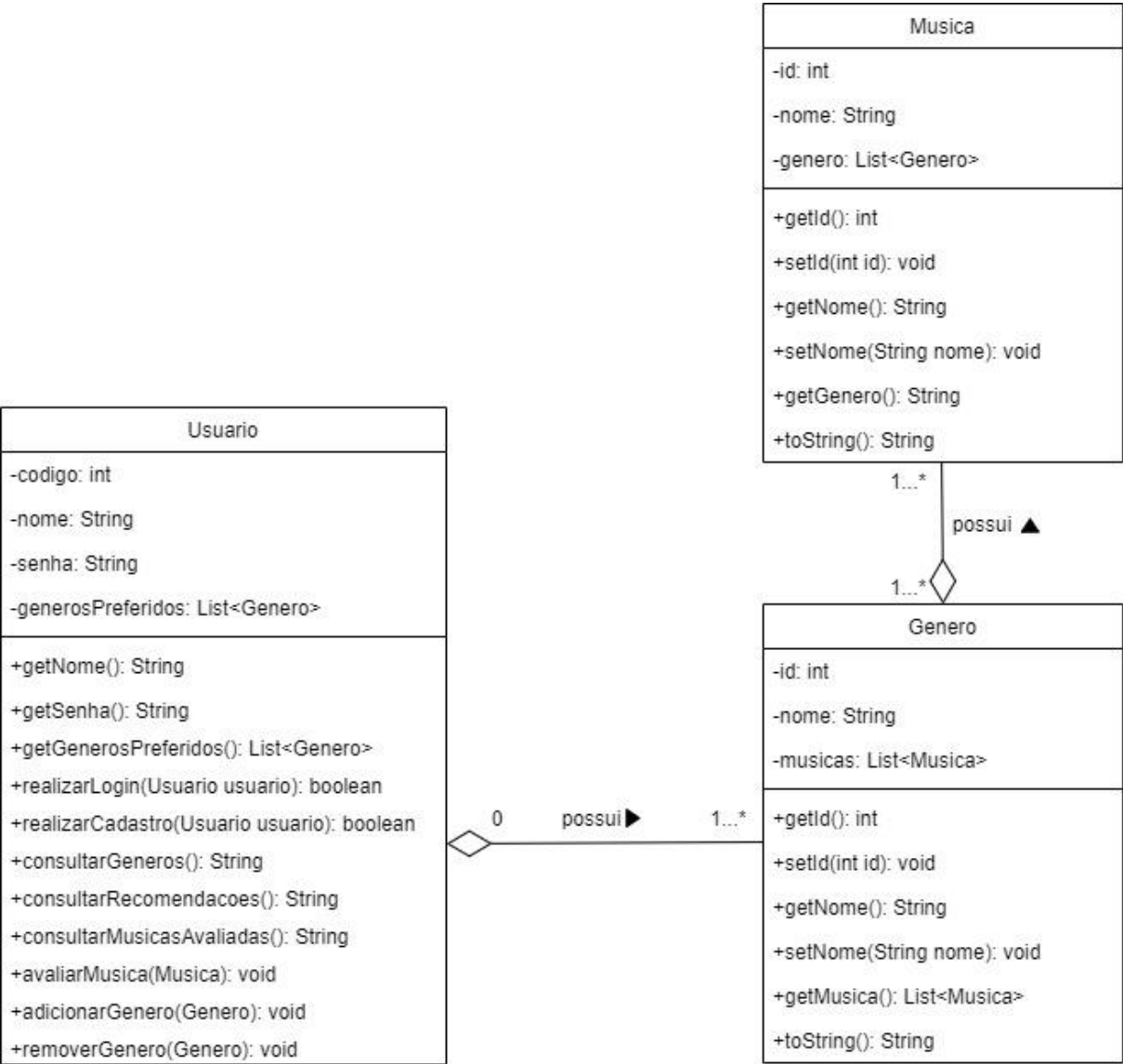
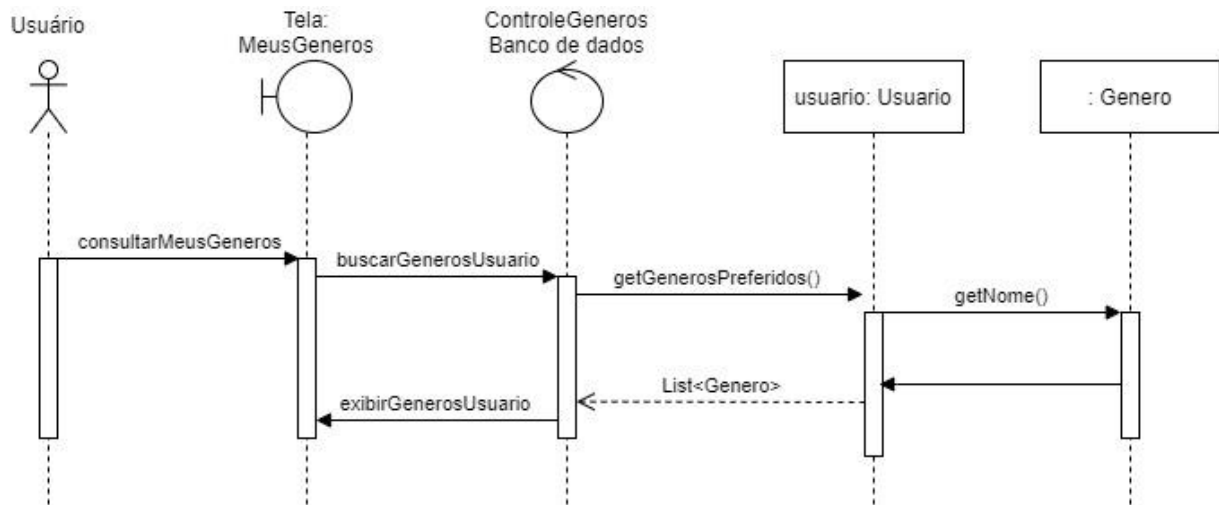


DIAGRAMA DE CLASSES



## DIAGRAMA DE SEQUÊNCIA

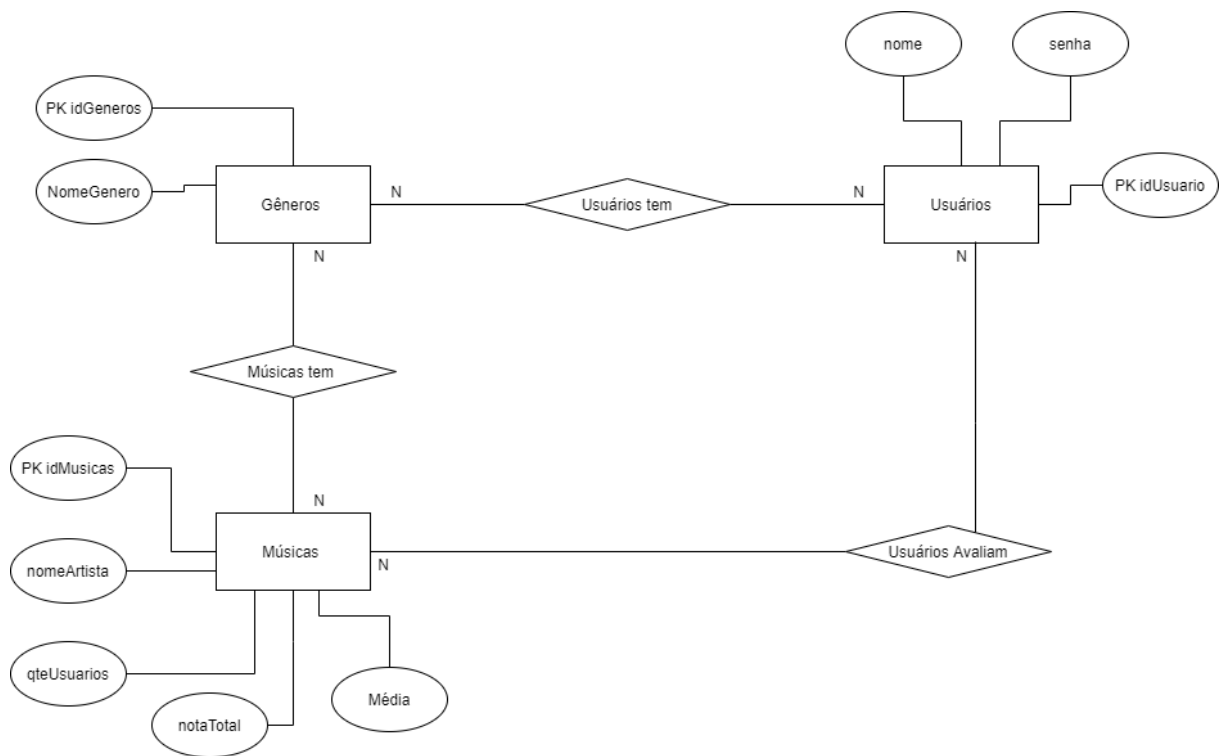
Atividade *core* do sistema: Consultar gêneros do Usuário



1. Usuário clica no painel “Meus Gêneros” da tela “Dashboard” para consultar os gêneros da sua coleção de gêneros
2. O sistema abre a tela “Meus Gêneros” para o usuário ver sua coleção de gêneros exibida na tela
3. Sistema busca cada um dos gêneros da coleção do usuário que são representados pela classe Genero no sistema
4. Sistema acessa a coleção de gêneros (coleção de objetos da classe Genero) do usuário
5. Cada usuário é representado por um objeto da classe Usuario
6. Sistema acessa o método getNome() dos objetos da classe Genero, para acessar os nomes dos gêneros
7. Sistema recebe o retorno da coleção de gêneros do usuário, com os nomes dos gêneros

8. Sistema lista os nomes dos objetos da classe Genero para que a tela “Meus Gêneros” liste os nome dos gêneros para o usuário
9. O usuário consulta os gêneros de sua coleção de gêneros exibidos na tela “Meus Gêneros”

## MODELO CONCEITUAL ER DO BANCO DE DADOS



## MODELO LÓGICO RELACIONAL DO BANCO DE DADOS

