

# Sistema de Música Suite Lite

Guilherme Sarturi, Murilo Thomaselli Teichert

Engenharia de Software/Sistemas de Informação  
Universidade da Região de Joinville (UNIVILLE) – Joinville, SC – Brazil

[guilherme.sarturi@univille.br](mailto:guilherme.sarturi@univille.br), [murilo.teichert@univille.br](mailto:murilo.teichert@univille.br)

## 1. Introdução

O Suite Lite é uma plataforma de streaming de música desenvolvida para proporcionar uma experiência envolvente e intuitiva, similar ao Spotify. Com foco na descoberta de novos gêneros, na organização de playlists e na interação social entre os usuários, o Suite Lite permite que os amantes da música explorem, sigam artistas e personalizem sua experiência de audição de maneira prática e acessível. Esta plataforma busca não apenas atender às necessidades dos usuários, mas também enriquecer sua jornada musical.

## 2. Requisitos Funcionais

Os requisitos funcionais do Sistema XPO serão apresentados em forma de história de usuário.

### 2.1. História de Usuário 01

Quero poder adicionar músicas em uma playlist!

Essa abordagem centrada no usuário ajuda a estabelecer a funcionalidade essencial de gerenciamento de playlists.

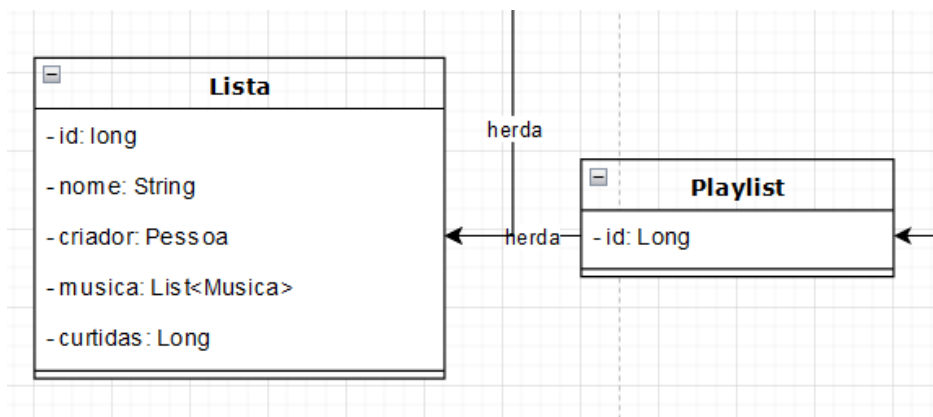


Figura 1. Modelo diagrama de classe da História de Usuário 01.

## 2.2. História de Usuário 02

Quero poder seguir meus artistas/amigos

Essa perspectiva do usuário possibilita que o sistema proporcione uma experiência mais personalizada, mantendo os usuários próximos aos seus artistas preferidos e seus amigos.

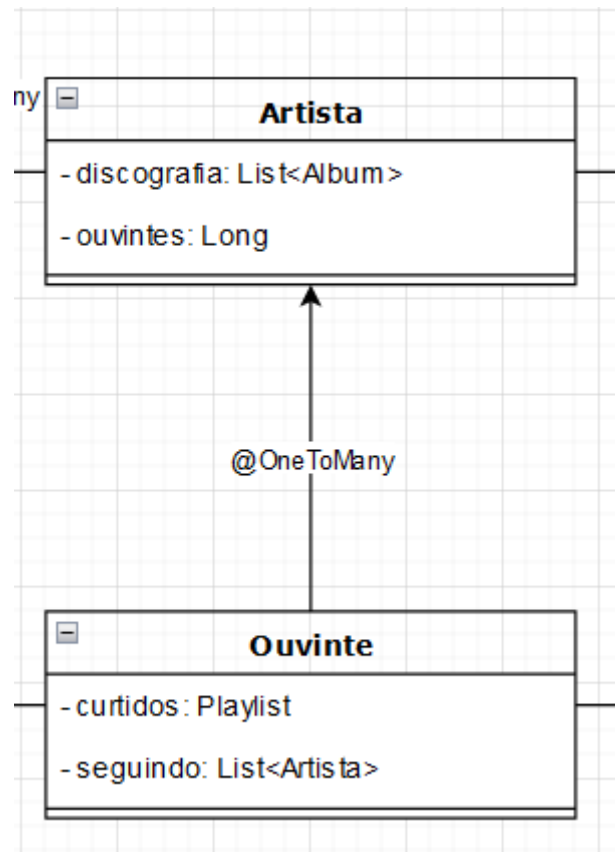


Figura 2. Modelo Entidade Relacionamento da História de Usuário 02.

## 2.3. História de Usuário 03

Artista: quero poder adicionar minhas músicas na plataforma

Essa abordagem centrada no usuário assegura que os artistas consigam compartilhar suas músicas com o público, o que facilita o desenvolvimento de suas carreiras e a promoção de seu conteúdo musical.

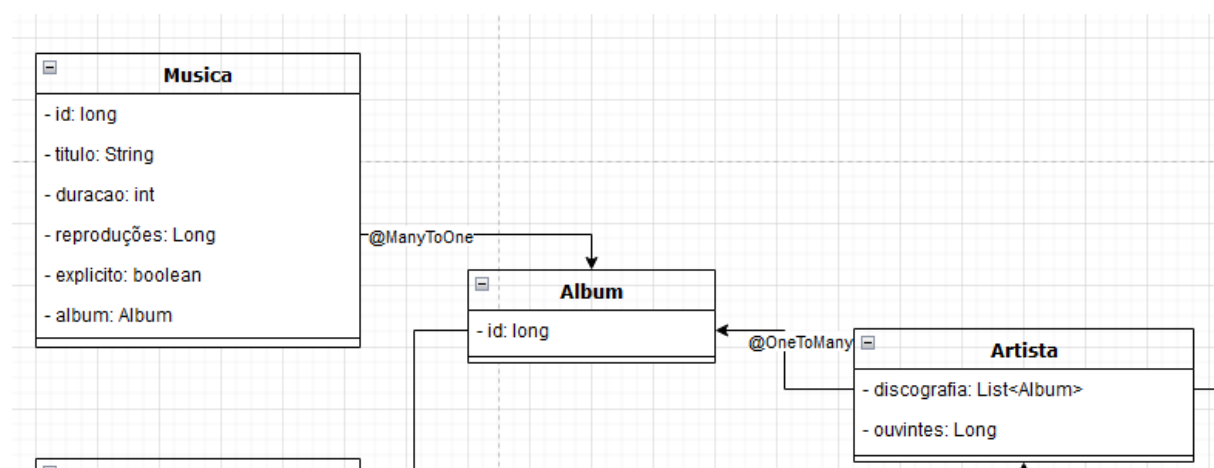


**Figura 3. Modelo Entidade Relacionamento da História de Usuário 03.**

## 2.4. História de Usuário 04

Artista: quero atribuir músicas a álbuns

Essa abordagem do usuário auxilia o artista a organizar suas músicas na plataforma, oferecendo uma experiência clara e intuitiva para os fãs.



**Figura 4. Modelo Entidade Relacionamento da História de Usuário 04.**

## 2.5. História de Usuário 05

Quero colocar músicas em playlists

Essa abordagem do usuário torna uma melhor experiência ao usar o app, por oferecer mais opções de personalização na forma de ouvir músicas.

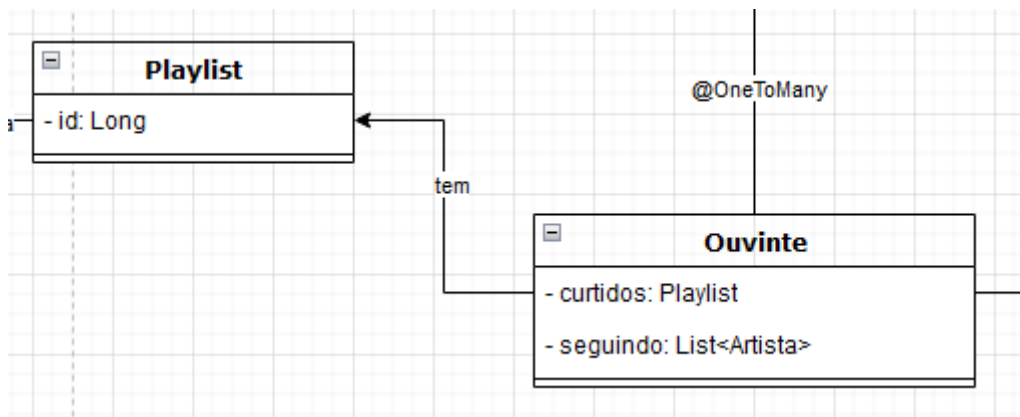


Figura 6. Modelo Entidade Relacionamento da História de Usuário 05.

### 3. Codificação

Apresentar as entidades e como realizou os relacionamentos. Apresentar o Diagrama completo em forma de figura 4.

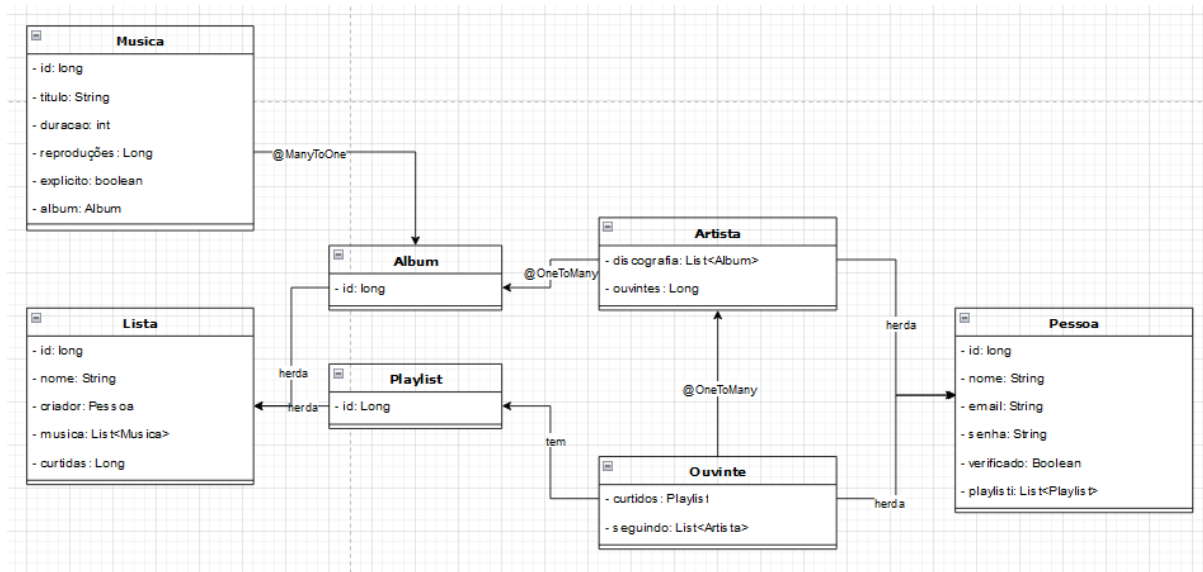


Figura 4. Diagrama de classe do Sistema de música 4

### 3.1. Entidade XOP

Album:

```
1  package br.edu.univille.br.relacionamentos.entities;
2
3  import jakarta.persistence.*;
4  import lombok.Data;
5  import lombok.EqualsAndHashCode;
6  import lombok.NoArgsConstructor;
7
8  @Data
9  @EqualsAndHashCode(callSuper = true)
10 @NoArgsConstructor
11 @Entity
12  public class Album extends Lista{
13      @Id
14      @GeneratedValue(strategy = GenerationType.IDENTITY)
15      private long id;
16  }
```

Figura 5. Entidade Album

Entidade Artista:

```
1  package br.edu.univille.br.relacionamentos.entities;
2
3  import java.util.List;
4
5  import jakarta.persistence.*;
6  import lombok.Data;
7  import lombok.EqualsAndHashCode;
8  import lombok.NoArgsConstructor;
9
10 @Data
11 @EqualsAndHashCode(callSuper = true)
12 @NoArgsConstructor
13 @Entity
14
15 public class Artista extends Pessoa{
16     @Id
17     @GeneratedValue(strategy = GenerationType.IDENTITY)
18     private long id;
19
20     private long ouvintes;
21
22     @OneToMany
23     @JoinColumn(name = "artista_id")
24     private List<Album> discografia;
25 }
```

Figura 6: Entidade Artista

Entidade Lista:

```
1  package br.edu.univille.br.relacionamentos.entities;
2
3  import java.util.List;
4
5  import jakarta.persistence.*;
6  import lombok.Data;
7  import lombok.NoArgsConstructor;
8
9  @Data
10 @NoArgsConstructor
11 @Entity
12
13 public class Lista {
14     @Id
15     @GeneratedValue(strategy = GenerationType.IDENTITY)
16     private long id;
17
18     @Column(nullable = false)
19     private String nome;
20
21     @ManyToOne
22     @JoinColumn(name = "criador_id")
23     private Pessoa criador;
24
25     @OneToMany
26     @JoinColumn(name = "lista_id")
27     private List<Musica> musicas;
28
29     private long curtidas;
30 }
```

Figura 8: Entidade Lista

Entidade Musica:

```
1 package br.edu.univille.br.relacionamentos.entities;
2
3 import jakarta.persistence.*;
4 import lombok.Data;
5 import lombok.NoArgsConstructor;
6
7 @Data
8 @NoArgsConstructor
9 @Entity
10
11 public class Musica {
12     @Id
13     @GeneratedValue(strategy = GenerationType.IDENTITY)
14     private long id;
15
16     private String nome;
17
18     @ManyToOne
19     @JoinColumn(name = "album_id")
20     private Album album;
21     private int duracao;
22     private long reproducoes;
23     private boolean explicito;
24 }
```

Figura 9: Entidade Música



Entidade Ouvinte:

```
1 package br.edu.univille.br.relacionamentos.entities;
2
3 import java.util.List;
4
5 import jakarta.persistence.*;
6 import lombok.Data;
7 import lombok.EqualsAndHashCode;
8 import lombok.NoArgsConstructor;
9
10 @Data
11 @EqualsAndHashCode(callSuper = true)
12 @NoArgsConstructor
13 @Entity
14
15 public class Ouvinte extends Pessoa{
16     @Id
17     @GeneratedValue(strategy = GenerationType.IDENTITY)
18     private long id;
19
20     @OneToOne
21     @JoinColumn(name = "curtidos_id")
22     private Playlist curtidos;
23
24     @OneToMany
25     @JoinColumn(name = "ouvinte_id")
26     private List<Artista> seguindo;
27 }
```

Figura 10: Entidade Ouvinte

Entidade Pessoa:

```

1  package br.edu.univille.br.relacionamentos.entities;
2
3  import java.util.List;
4
5  import jakarta.persistence.*;
6
7  import jakarta.persistence.*;
8  import lombok.Data;
9  import lombok.NoArgsConstructor;
10
11  @Data
12  @NoArgsConstructor
13  @Entity
14
15  ✓ public class Pessoa {
16      @Id
17      @GeneratedValue(strategy = GenerationType.IDENTITY)
18      private long id;
19
20      @Column(nullable = false)
21      private String nome;
22
23      @Column(nullable = false)
24      private String senha;
25
26      @Column(nullable = false)
27      private String email;
28
29      private boolean verificado;
30
31      @OneToMany
32      @JoinColumn(name = "pessoa_id")
33      private List<Playlist> playlists;
34  }

```

Figura 12: Entidade Pessoa

Entidade Playlist:

```
1    package br.edu.univille.br.relacionamentos.entities;
2
3    import jakarta.persistence.*;
4    import lombok.Data;
5    import lombok.EqualsAndHashCode;
6    import lombok.NoArgsConstructor;
7
8    @Data
9    @EqualsAndHashCode(callSuper = true)
10   @NoArgsConstructor
11   @Entity
12
13   public class Playlist extends Lista{
14       @Id
15       @GeneratedValue(strategy = GenerationType.IDENTITY)
16       private long id;
17   }
```

Figura 13: Entidade Playlist

#### 4. Banco de dados

Diagrama MER conforme estrutura das entidades:

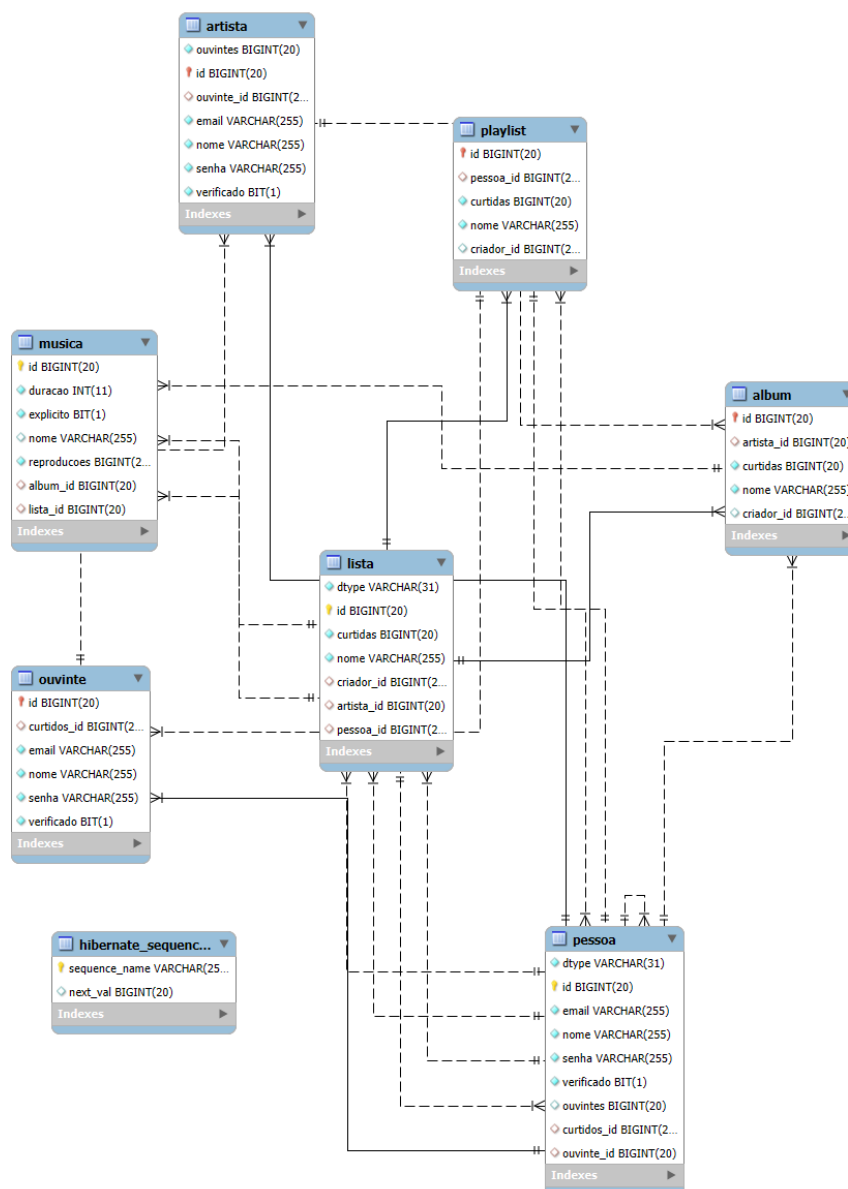


Figura 2. Modelo Entidade Relacionamento do Suite Lite

#### 4. Conclusão

Em suma, o Suite Lite se destaca como uma solução completa para os amantes da música, oferecendo ferramentas que não apenas facilitam a exploração de novos sons e artistas, mas também promovem uma conexão significativa entre usuários. Ao combinar uma interface intuitiva com funcionalidades sociais e de personalização, o Suite Lite está comprometido em transformar cada momento de audição em uma experiência única e envolvente. Prepare-se para mergulhar em um mundo musical que se adapta aos seus gostos e preferências.