

1. Visão Geral do Projeto

Desenvolver uma plataforma online location-based-service (LBS) cujo objetivo é facilitar o encontro, comunicação e colaboração entre músicos. O propósito central da aplicação é conectar pessoas que compartilham gostos musicais semelhantes, possam se conectar uma com a outra e colaborar de alguma maneira.

A plataforma visa atender tanto músicos iniciantes que procuram aprendizado e experiências, quanto profissionais em busca de colaborações mais estruturadas.

Principais objetivos:

- Conectar músicos próximos geograficamente.
- Permitir colaboração musical remota e presencial.
- Estimular a formação de comunidades artísticas locais.

Atende a ODS 17 - Parcerias e Meios de Implementação. Por fomentar a conexão entre músicos, bandas, escolas de músicas, ONGs culturais promovendo parcerias e estimulando a indústria musical na sociedade.

2. Requisitos funcionais

[RF001] Cadastro e Autenticação

Descrição: O sistema deve permitir criação de conta com email, nome de usuário e senha.

Atores: User

Critérios de Aceitação:

- O usuário deve fornecer um Email válido para criar seu perfil.
- A senha deve ter critérios mínimos (8 caracteres, número, letra maiúscula).
- Login e logout devem estar disponíveis em qualquer dispositivo.
- Recuperação de senha via email deve ser implementada.

[RF002] Gerenciar Perfil Artístico

Descrição: O usuário deve poder criar e gerenciar seu perfil com atributos como Nome, Biografia, Instrumentos, Gêneros Musicais, Objetivo (formar banda, jam, networking etc.), links externos (SoundCloud, Spotify, YouTube) e portfólio multimídia.

Atores: User

Critérios de Aceitação:

- Perfil deve permitir atualização a qualquer momento.
- Links externos devem ser validados.
- Upload de imagens (foto de perfil e capa).
- Suporte para upload de áudio/vídeo em formatos comuns (mp3, wav, mp4, mov).

[RF003] Sistema de Descoberta

Descrição: O sistema deve permitir busca de outros músicos com base em localização e filtros personalizados.

Atores: User

Critérios de Aceitação:

- Busca por raio geográfico com limite configurável (ex: até 50 km).
- Filtros por cidade, estado, gênero musical, instrumento e objetivo.
- Exibir distância aproximada em km.

[RF004] Interface de Descoberta por Cards

Descrição: O sistema deve apresentar potenciais matches um de cada vez em um formato de card interativo. O usuário poderá deslizar o card para a direita para dar "like" (curtir) ou para a esquerda para "passar" (descartar).

Atores: User

Critérios de Aceitação:

- Os cards devem exibir as informações essenciais do perfil: foto, nome, instrumentos, gêneros, distância e uma breve bio.
- A animação do card sendo descartado deve ser fluida e dar feedback claro da ação.
- Os gestos de swipe (arrastar) devem ser suportados, além de botões claros para "Curtir" e "Passar".
- Ao esgotar os perfis na área geográfica, o sistema deve informar o usuário.
- Exibir distância aproximada em km.

[RF005] Feed de Postagens

Descrição: Permite aos usuários postarem clips, convites para jam e colaboração. Essas postagem podem ter curtidas e comentários.

Atores: User

Critérios de Aceitação:

- Permitir postagens de texto, áudio, imagem ou vídeo.
- Curtidas e comentários em cada postagem.
- Opção de denúncia para conteúdo inadequado.
- Feed ordenado por relevância ou cronologia.

[RF006] Sistema de Chat

Descrição: Usuários devem ser capazes de enviar mensagens para qualquer outro usuário, com possibilidade de enviar áudios curtos, arquivos e links. A mensagem inicial deve servir como um pedido de conversa ou colaboração.

Atores: User

Critérios de Aceitação:

- Mensagens em texto, envio de áudios curtos, imagens, links e arquivos
- Possibilidade de bloquear usuários indesejados.
- Possibilidade de bloquear usuários indesejados. Ao bloquear, as mensagens existentes são arquivadas e novas mensagens são impedidas.
- Suporte para notificações

[RF007] Notificação

Descrição: O sistema deve enviar notificações para interações relevantes.

Atores: User

Critérios de Aceitação:

- Enviar notificação em caso de nova mensagem recebida.
- Novo seguidor.
- Curtida/comentário em postagem.
- Solicitação de colaboração, referente ao sistema de colaboração.

[RF008] Moderação e Segurança

Descrição: Garantir um ambiente seguro para interação entre usuários.

Atores: User, Administrador

Critérios de Aceitação:

- Possibilidade de denúncia (User) e exclusão (Adm) de usuários e conteúdos.
- Moderação automática para palavras ofensivas.
- Bloqueio temporário ou definitivo de contas reincidentes.
- Termos de uso e política de privacidade claros e acessíveis..

3. Requisitos Não Funcionais

[RNF001] Cálculo de Distância

Descrição: O sistema deve calcular e exibir a distância aproximada entre usuários (em km), garantindo boa performance mesmo em cenários de alta demanda.

[RNF002] Usabilidade

Descrição: Interface intuitiva, com design responsivo para mobile e desktop.

[RNF003] Disponibilidade

Descrição: O sistema deve ter alta disponibilidade.

[RNF004] Escalabilidade

Descrição: O sistema deve ser capaz de suportar pelo menos 100 usuários simultâneos em sua primeira release.

4. Casos de Uso

UC001 – Criar Conta

Atores: User

Fluxo Principal:

1. O usuário acessa a tela de cadastro.
2. Informa e-mail, nome de usuário e senha.
3. O sistema valida e-mail e senha.

4. O sistema registra o usuário e envia e-mail de confirmação.

Fluxo Alternativo:

- Se o e-mail já estiver cadastrado, o sistema deve exibir mensagem de erro.

UC002 – Criar Perfil Artístico

Atores: User

Fluxo Principal:

1. O usuário insere dados pessoais
2. O usuário seleciona instrumentos, gêneros, artistas favoritos, objetivo na plataforma e links.
3. Perfil salvo e visível dentro da plataforma

Fluxo Alternativo:

- O usuário, depois da inserção dos dados pessoais, pode optar por pular o processo inicial e inserir os dados mais tarde.

UC003 – Descoberta de Músicos

Atores: User

Fluxo Principal:

1. O usuário acessa tela de busca
2. O usuário define raio e outros filtros
3. O usuário recebe uma lista de cards empilhados exibindo dados básicos de outros usuários.
4. O usuário pode deslizar os cards para esquerda (passar para o próximo) ou deslizar o card para direita e enviar um pedido de conexão e uma mensagem para o outro usuário.

Fluxo Alternativo:

- O usuário pode acessar e visualizar o perfil do usuário pelo card exibido.

UC004 – Pesquisar Perfis no Chat de Perfis Descobertos

Atores: User

Fluxo Principal:

1. O usuário acessa a barra de pesquisas na interface de chat dos “perfis descobertos”
2. O usuário insere uma query para buscar por outros usuários por username
3. O usuário recebe uma lista com os usuários com os filtros aplicados

UC005 – Pesquisar Posts no Feed

Atores: User

Fluxo Principal:

1. O usuário acessa a barra de pesquisas na tela de Feed para procurar algum post no seu histórico Postagens recomendadas
2. O usuário insere uma query para buscar por outros usuários por username, ou palavras-chaves baseadas nas tags do post de dentro do feed ou até mesmo pelo título do post
3. O usuário recebe uma lista com os posts com os filtros aplicados

UC006 – Iniciar Chat

Atores: User

Fluxo Principal:

1. O usuário X acessa o perfil do usuário Y, seja pela descoberta ou pesquisa
2. O Usuário envia pedido de conexão junto de uma mensagem para Y
3. O usuário Y tem a opção de aceitar a conexão e o continuar o chat, ou negar a conexão e fechar o chat.

UC007 – Criar Postagem

Atores: User

Fluxo Principal:

1. O usuário acessa a tela de feed ou seu perfil
2. O usuário seleciona a opção "Criar Postagem"
3. O sistema exibe opções de tipo de conteúdo (texto, áudio, vídeo)
4. O usuário insere o conteúdo da postagem
5. O usuário pode adicionar tags e definir visibilidade (Público/Privado)
6. O usuário confirma a publicação
7. O sistema salva a postagem e a exibe no feed

Fluxo Alternativo:

- 3a. O usuário pode acessar e visualizar o perfil do usuário pelo card exibido.
- 5a. O usuário pode cancelar a postagem a qualquer momento

UC008 – Interagir com Postagem

Atores: User

Fluxo Principal:

1. O usuário visualiza uma postagem no feed
2. O usuário pode curtir, comentar ou compartilhar a postagem
3. O sistema registra a interação e atualiza os contadores
4. O autor da postagem recebe notificação da interação

Fluxo Alternativo:

- 2a. O usuário pode reportar a postagem como inadequada
- 2b. O usuário pode responder a comentários existentes

UC009 – Criar Colaboração

Atores: User

Fluxo Principal:

5. O usuário acessa a seção de colaborações
6. O usuário seleciona a opção "Criar Colaboração"
7. O sistema exibe formulário sobre os dados de criação do pedido de colaboração

8. O usuário preenche o formulário
9. O usuário seleciona outros participantes(se houverem)
10. O sistema cria a colaboração e notifica os participantes

Fluxo Alternativo:

- 5a. O usuário pode convidar participantes específicos de sua lista de conexões
- 6a. Colaborações podem ser editadas ou canceladas pelo usuário criador

5. Entidades do Domínio

User: {

id: Long,
email: String,
password: String,
username: String,
bio: String,

location: {
latitude: Double,
longitude: Double,
city: String,
state: String,
country: String
}

profilePicture: String(URL),
profileBanner: String(URL),
followers: List<User>,
skillLevel: Enum (BEGINNER, INTERMEDIATE, ADVANCED, PROFESSIONAL),
connections: List<User>,
instruments: List<String>,
favoriteArtists: List<String>
genres: List<String>,
goals: List<String>,
links: {spotify, soundcloud, youtube},
blockedUsers: List<User>,
emailVerified: Boolean,
mediaPortfolio: List<Post>
createdAt: Date

}

Post: {


```
id: Long,
authorId(FK): User,
content: caption,
mediaUrl: String(Youtube, spotify, soundcloud),
likes: int,
tags: List<String>,
comments: List<Comment>,
reported: Boolean,
reports: List<Report>,
created_at: Date,
updated_at: Date,
visibility: Enum (PUBLIC, PRIVATE),
}
```

```
Collaboration:: {
id: Long,
title: String,
description: String,
initiator: User,
participants: List<User>,
status: Enum (PLANNING, IN_PROGRESS, COMPLETED, CANCELLED),
type: Enum (JAM_SESSION, RECORDING, LIVE_PERFORMANCE, SONGWRITING),
scheduledDate: Date,
location: {
type: Enum (VIRTUAL, PHYSICAL),
address: String,
coordinates: { latitude, longitude }
},
chatChannel: Chat,
createdAt: Date,
updatedAt: Date
}
```

```
Report: {
id: Long,
reporter: User,
reportedEntity: {
type: Enum (USER, POST, COMMENT),
id: Long
},
reason: Enum (SPAM, HARASSMENT, INAPPROPRIATE_CONTENT, IMPERSONATION),
description: String,
status: Enum (PENDING, UNDER_REVIEW, RESOLVED, DISMISSED)
createdAt: Date,
updatedAt: Date
}
```

```
Comment: {
```

```
id: Long,
postId(FK): Post,
authorId(FK): User,
content: String,
createdAt: Date,
updatedAt: Date
}
```

```
Notification: {
  id: Long,
  userId: User,
  type: Enum (MESSAGE, LIKE, COMMENT, CONNECTION, COLLABORATION, SYSTEM),
  content: String,
  relatedEntity: {
    type: String,
    id: Long
  },
  isRead: Boolean,
  priority: Enum (LOW, MEDIUM, HIGH)
}
```

```
Chat: {
  id: Long,
  participants: List<User>,
  messages: List<Message>,
  createdAt: Date,
  updatedAt: Date
}
```

```
Message: {
  id: Long,
  chatId(FK): Chat,
  senderId(FK): User,
  messageType: Enum (TEXT, AUDIO, IMAGE, FILE, SYSTEM),
  content: String (texto),
  mediaUrl: String (imagem, áudio curto, arquivo),
  createdAt: Date,
  isRead: Boolean
}
```

6. Tecnologias

Frontend: React (web) + React Native (Mobile)

Backend: Spring Boot

Banco de Dados: PostgreSQL

Autenticação: JWT + OAuth2

API Geolocalização: OpenStreetMap

7. Telas

Landing page

Feed

Perfil Usuário

Listagem de Pesquisa

Tela de deslizamento de cards (coleção)

Tela de Cadastro