Logotipo, nome da empresa

Descrição gerada automaticamente

**FACULDADE DE TECNOLOGIA DO IPIRANGA**

**CURSO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

DEISE DE ALMEIDA DOMINGUES

GIOVANNA FONSECA DA SILVA

**Proposta Técnica**

Musician & Match

Plataforma para conectar músicos a eventos

**Orientador:** Prof. Rodrigo Bossini Tavares Moreira

**Professor da Disciplina:** Prof. Carlos Henrique Verissimo Pereira

SÃO PAULO

2024

Sumário

[Proposta Técnica – Musician & Match 3](#_Toc191408278)

[Visão geral do produto 3](#_Toc191408279)

[Definição das partes interessadas 3](#_Toc191408280)

[Especificação de personas 3](#_Toc191408281)

[Persona 1 – Contratante Casual 4](#_Toc191408282)

[Persona 2 – Música freelancer 4](#_Toc191408283)

[Persona 3 – Músico à procura de banda 5](#_Toc191408284)

[Modelagem SQL 5](#_Toc191408285)

[Diagramas UML 7](#_Toc191408286)

[Diagrama de caso de uso 7](#_Toc191408287)

[Diagrama de Classes 8](#_Toc191408288)

[Arquitetura de solução 8](#_Toc191408289)

[Ecossistema da Solução de Software 9](#_Toc191408290)

# Proposta Técnica – Musician & Match

## Visão geral do produto

O **Musician & Match** é um aplicativo móvel desenvolvido para conectar contratantes a músicos profissionais, simplificando a busca por instrumentistas para eventos de diversos tipos. Com uma interface intuitiva e organizada, a plataforma oferece filtros por região e data, além de exibir informações de contato dos músicos e um sistema de avaliação pós-evento, garantindo uma experiência eficiente e acessível para os usuários.

Diferente de plataformas que intermediam contratações e pagamentos, o **Musician & Match** atua apenas como um facilitador, permitindo que contratantes encontrem músicos disponíveis para a data desejada e entrem em contato diretamente para negociações externas. Seja para apresentações solo ou substituições em bandas, o aplicativo atende tanto a demandas informais, como um violonista para um jantar, quanto a eventos menores que normalmente não são atendidos por empresas especializadas.

A dificuldade em encontrar músicos disponíveis, especialmente para eventos de menor porte, é um desafio frequente. Muitas soluções existentes são limitadas a redes de contatos restritas ou a agências que cobram altos valores. O **Musician & Match** surge como uma alternativa inovadora, permitindo que os próprios músicos ofereçam seus serviços de forma autônoma, sem intermediários. Dessa forma, a plataforma torna o processo mais ágil, acessível e personalizado, conectando profissionais qualificados a contratantes com eficiência.

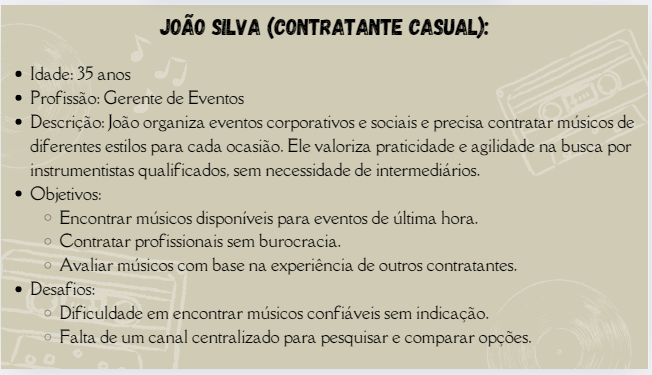
## Definição das partes interessadas

No projeto **Musician & Match**, as partes interessadas são os principais usuários e envolvidos no funcionamento da plataforma. São eles:

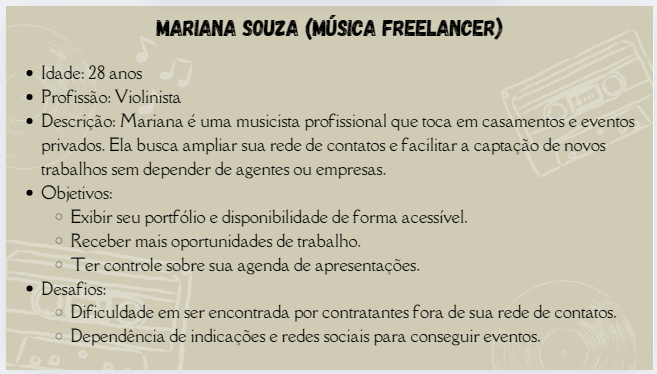
* **Músicos**: Profissionais que desejam divulgar seus serviços e se conectar com contratantes. Eles podem cadastrar seu perfil, listar instrumentos que tocam, definir disponibilidade e receber contatos para negociações diretas.
* **Contratantes**: Usuários que buscam músicos para eventos e apresentações. Eles podem pesquisar por instrumentistas, aplicar filtros por localização e disponibilidade e entrar em contato diretamente com os músicos para negociação.
* **Administradores da Plataforma**: Responsáveis por gerenciar e manter o funcionamento do aplicativo, garantindo que as funcionalidades operem corretamente e que a plataforma seja segura e confiável.

## Especificação de personas

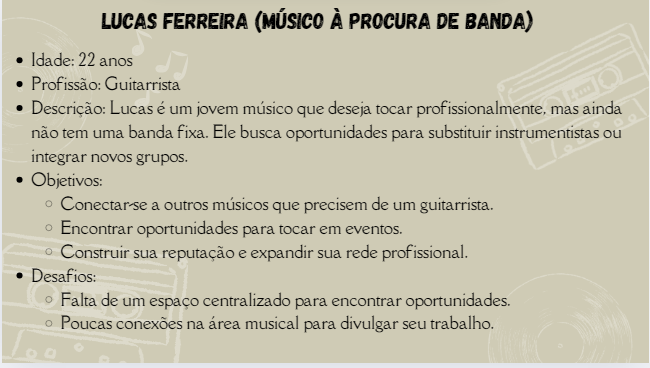
### **Persona 1 – Contratante Casual**



### **Persona 2 – Música freelancer**

****

### **Persona 3 – Músico à procura de banda**

****

## Modelagem SQL

CREATE TABLE USUARIO (

CPF VARCHAR(11) PRIMARY KEY,

NOME VARCHAR(100) NOT NULL,

EMAIL VARCHAR(100) NOT NULL,

TELEFONE VARCHAR(15),

TIPO ENUM('COMUM', 'MUSICO') NOT NULL

);

CREATE TABLE INSTRUMENTO (

ID INT AUTO\_INCREMENT PRIMARY KEY,

NOME VARCHAR(100) NOT NULL UNIQUE

);

CREATE TABLE MUSICO\_INSTRUMENTO (

CPF\_MUSICO VARCHAR(11),

ID\_INSTRUMENTO INT,

PRIMARY KEY (CPF\_MUSICO, ID\_INSTRUMENTO),

FOREIGN KEY (CPF\_MUSICO) REFERENCES USUARIO(CPF),

FOREIGN KEY (ID\_INSTRUMENTO) REFERENCES INSTRUMENTO(ID)

);

CREATE TABLE DISPONIBILIDADE (

ID INT AUTO\_INCREMENT PRIMARY KEY,

CPF\_MUSICO VARCHAR(11),

ID\_INSTRUMENTO INT,

DATA\_INICIO DATE NOT NULL,

DATA\_FIM DATE NOT NULL,

HORARIO\_INICIO TIME NOT NULL,

HORARIO\_FIM TIME NOT NULL,

FOREIGN KEY (CPF\_MUSICO) REFERENCES USUARIO(CPF),

FOREIGN KEY (ID\_INSTRUMENTO) REFERENCES INSTRUMENTO(ID),

CHECK (DATA\_INICIO <= DATA\_FIM),

CHECK (HORARIO\_INICIO < HORARIO\_FIM)

);

CREATE TABLE AVALIACAO (

ID INT AUTO\_INCREMENT PRIMARY KEY,

CPF\_USUARIO\_COMUM VARCHAR(11),

CPF\_MUSICO VARCHAR(11),

NOTA INTEGER CHECK (NOTA BETWEEN 0 AND 5),

COMENTARIO VARCHAR(500),

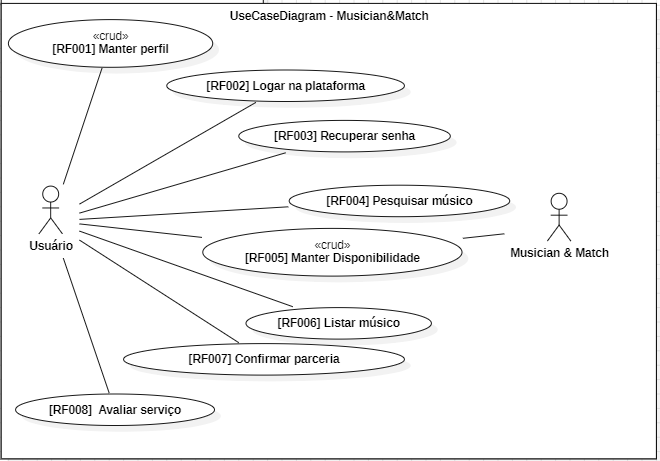
FOREIGN KEY (CPF\_USUARIO\_COMUM) REFERENCES USUARIO(CPF),

FOREIGN KEY (CPF\_MUSICO) REFERENCES USUARIO(CPF)

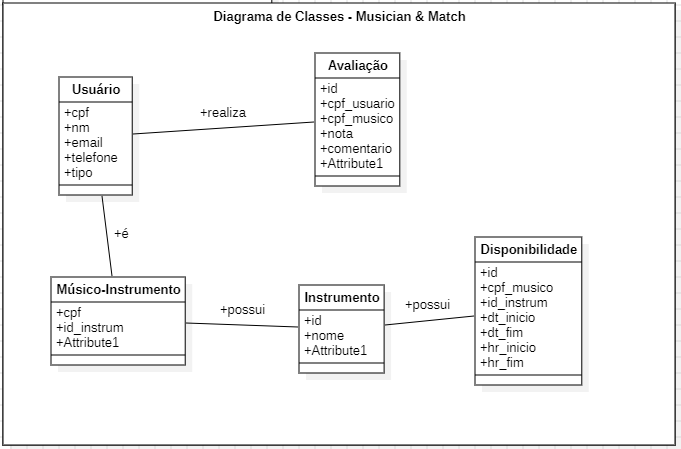
);

## Diagramas UML

### Diagrama de caso de uso

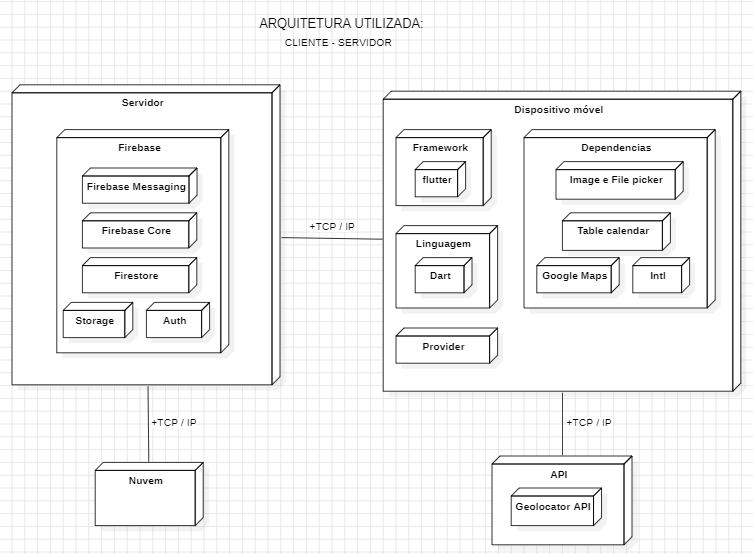
**

### Diagrama de Classes



## Arquitetura de solução

(Desatualizada – Alterar linguagem de programação)



## Ecossistema da Solução de Software

