

UNIFSA - Centro Universitário Santo Agostinho

Curso: Engenharia de Software
Disciplina: Análise e Projeto de Software
Professor: Anderson Soares Costa
Período Letivo: 2025.2

Trabalho de Composição da 2ª Nota

Tema: Modelagem de Sistema - Rede social para músicos independentes

Equipe:

- **Ângelo Gabriel da Silva Assis - Matrícula: XXXXXXXXXX**

Data da entrega: 24/10/2025

1. Descrição do Sistema

O sistema proposto é um mecanismo de conexão entre músicos independentes e seus fãs, propondo uma maneira do público apoiar os artistas diretamente por meio da compra de álbuns (digitais ou físicos) e de itens de *merchandising*, ou doação direta. A motivação principal do projeto é a ideia de fornecer ao pequeno artista uma fonte de renda mais estável, tendo em vista que as grandes plataformas de streaming não fornecem lucro suficiente na maioria dos casos.

Dentro do sistema, é possível criar uma conta como usuário comum ou artista, sendo necessário, no segundo caso, fazer upload de suas músicas e configurar a forma de monetização preferida. Por conta da plataforma, serão fornecidos mecanismos de divulgação, como uma página de busca por artistas locais e um *ranking* de “audiência”, ou seja, os artistas mais engajados com o público poderão ganhar um destaque. Para o fã, os benefícios consistirão no contato mais próximo com o músico, sendo possível enviar mensagens (mediante doação) e realizar a compra de artigos. O usuário também pode criar ou entrar em comunidades de fãs para interação.

Quando realizada a compra de um álbum ou música no formato digital, será possível fazer o download dos arquivos (preferencialmente no formato FLAC para maior definição) ou escutar pelo próprio sistema. Assim, uma funcionalidade de player deve ser bem implementada. Para o caso dos álbuns físicos, o fornecimento deve ser realizado por um *marketplace* de terceiros integrado, para maior segurança ao usuário.

2. Análise de requisitos

Como requisitos funcionais, pode-se listar:

- O usuário deve poder criar uma conta como fã ou artista;
- O fã poderá contribuir com artistas de diferentes formas, que devem ser suportadas pela plataforma por meio de um sistema seguro de vendas;

- O artista pode fazer *upload* de seus trabalhos e fornecê-los de forma gratuita e/ou paga;
- O fã, ao apoiar financeiramente o artista, pode enviar uma mensagem;
- Todos devem visualizar um ranking de artistas, classificados por diferentes critérios;
- O sistema deve fornecer ao usuário os arquivos digitais;
- O sistema deve possibilitar a escuta de áudio no formato FLAC;
- Deve ser integrado um sistema de vendas de terceiro (para itens físicos);
- Um mecanismo de comunidades deve ser implementado, para interação entre diferentes usuários

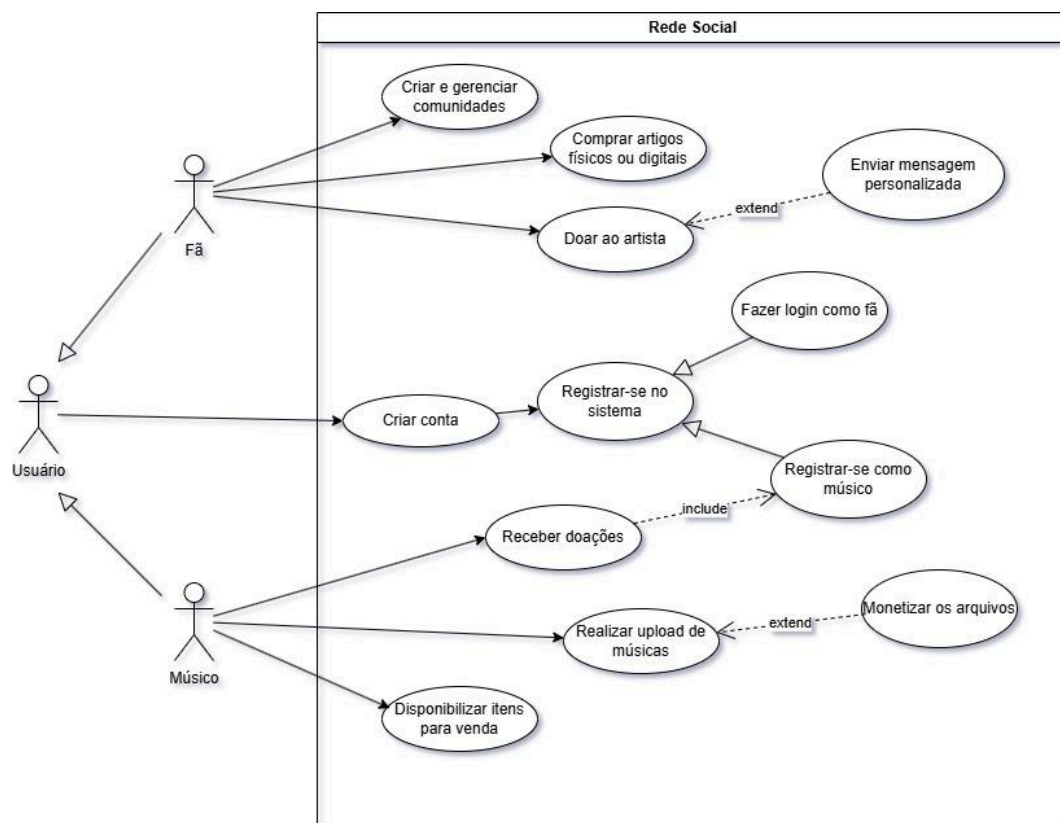
Como requisitos não-funcionais:

- O sistema deve suportar diferentes formatos de áudio, no mínimo dois, sendo um para visualização gratuita e um para os downloads;
- A interface deve ser intuitiva ao ponto de não ser um impeditivo para que ocorra o apoio dos fãs e a interação;
- Deve ser possível a utilização por meio de redes móveis;
- O tamanho total do app deve ser razoável, pensando em diferentes dispositivos e realidades;

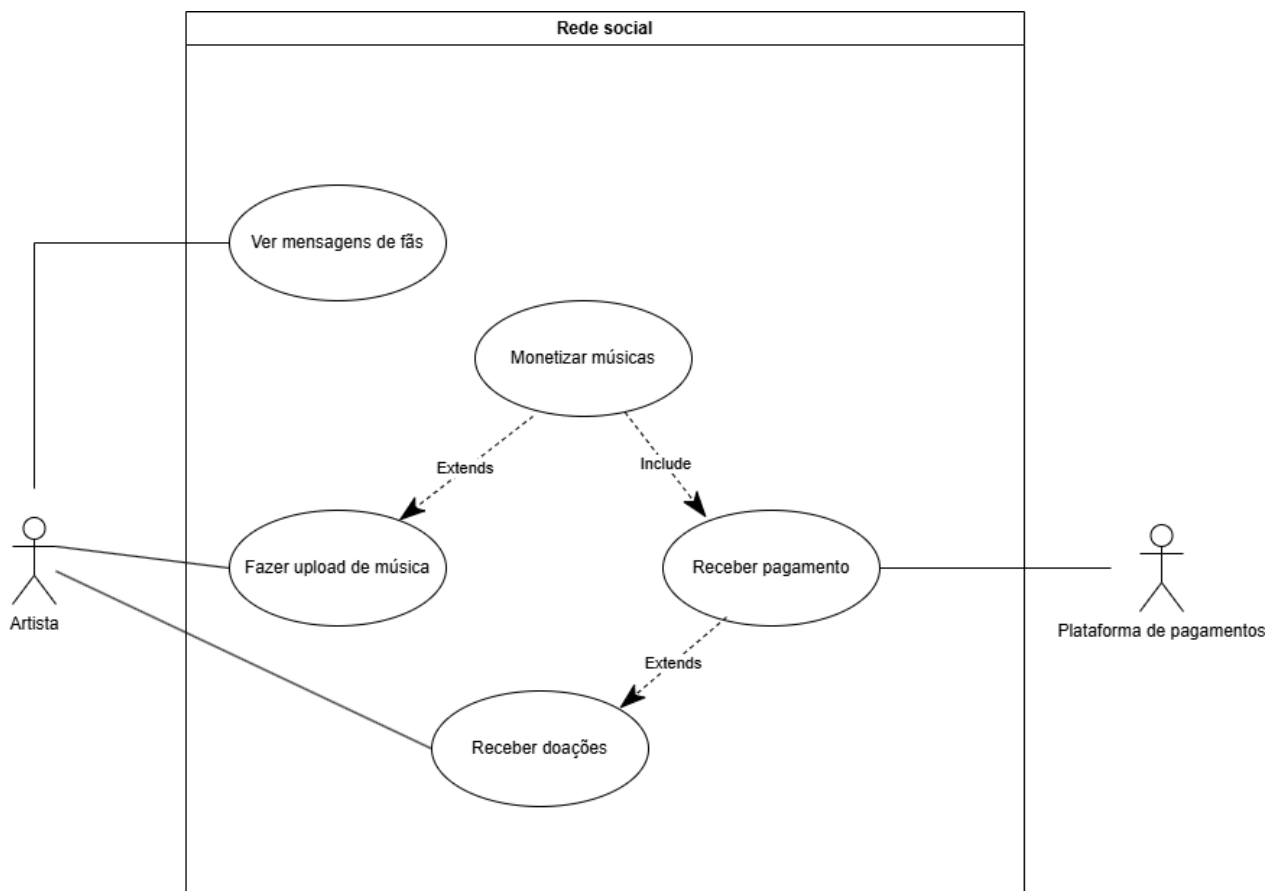
Como regras de negócio:

- Não deve haver taxas sobre as doações diretas;
- Um usuário não pode ser fã e artista simultaneamente, para evitar fraudes;
- Plataformas de terceiros empregadas devem ser seguras e confiáveis;

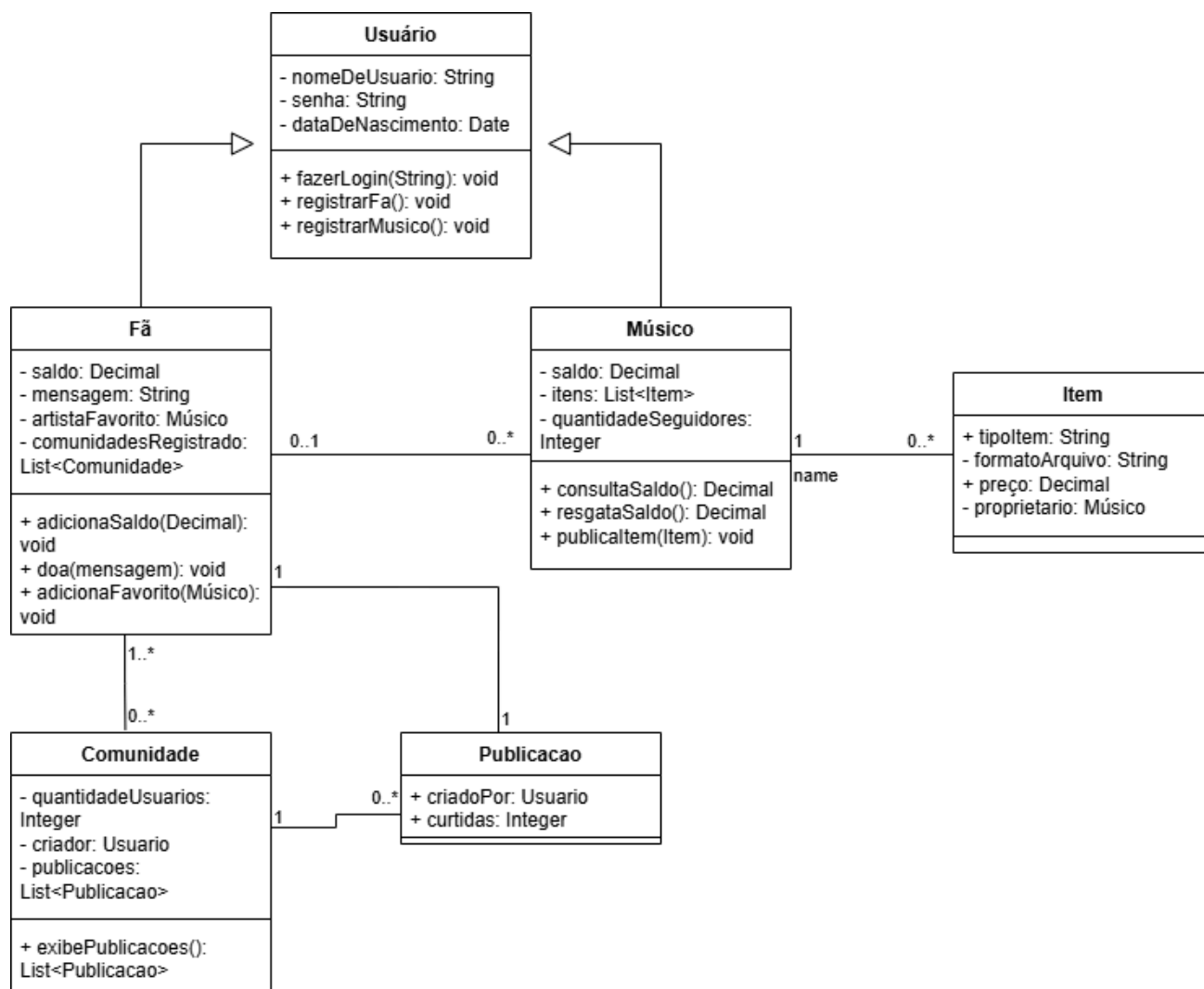
3. Modelagem com UML



O primeiro diagrama de caso de uso mostra o funcionamento geral da aplicação, detalhando como os usuários podem interagir com a plataforma como fã ou artista, e separando o que cada “classe” pode realizar no sistema.

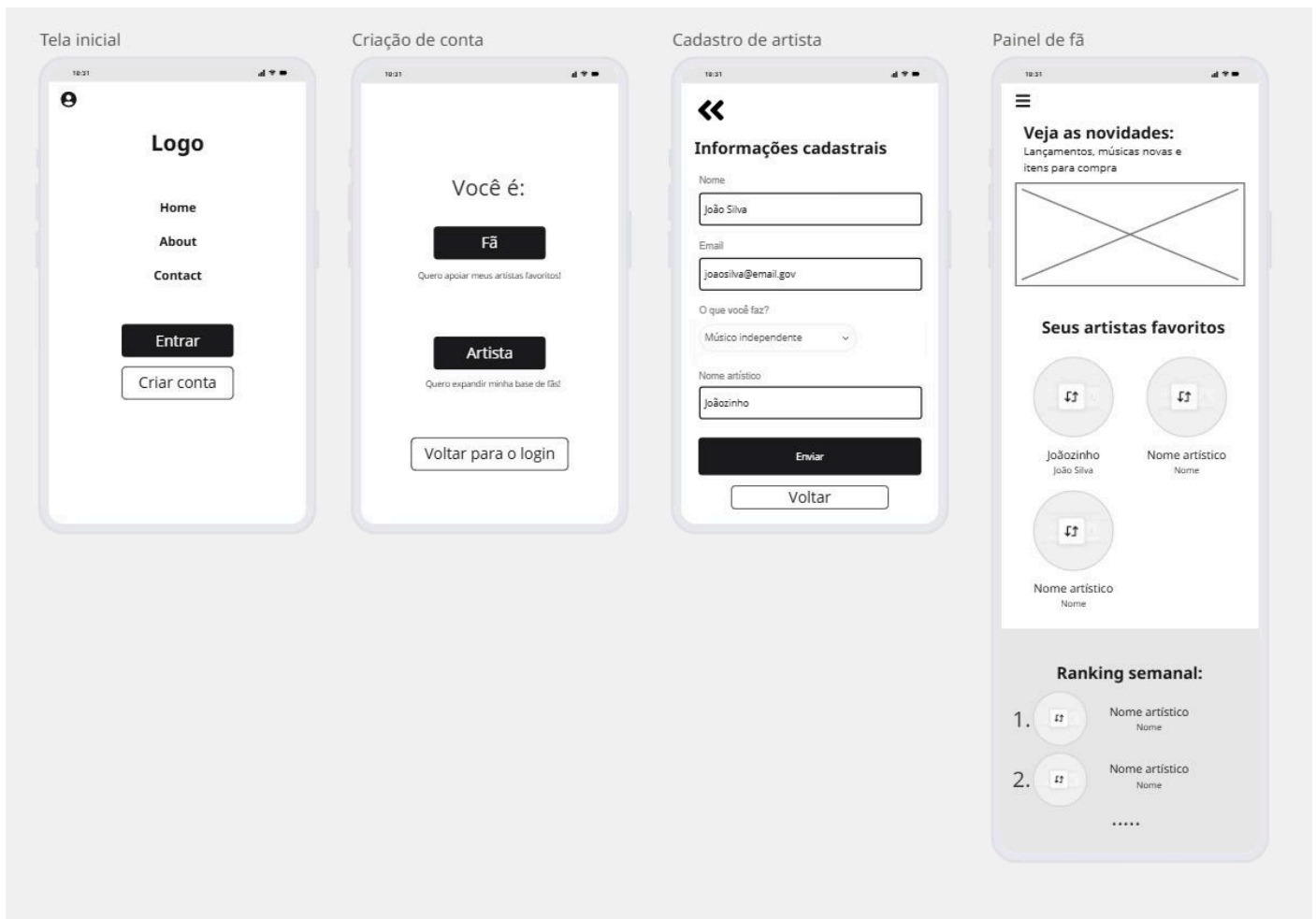


O segundo diagrama é focado na interação do músico com o sistema, e representa como a plataforma de pagamentos terceirizada pode agir.



Por fim, o diagrama de classes, que ilustra como as classes devem ser implementadas no sistema, destacando os relacionamentos entre elas e a cardinalidade. Além disso, o diagrama de classes possui os atributos e métodos de cada classe, com seus devidos tipos de dados.

4. Protótipos e wireframes



5. Link para o vídeo explicativo

6. Link para o Github

https://github.com/angelo-gabriel/aps_trabalho_av2