



INSTITUTO FEDERAL DO PIAUÍ – IFPI
PRÓ-REITORIA DE ENSINO
DIRETORIA DE ENSINO
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS

RELATÓRIO TÉCNICO DE SOFTWARE
MARKETPLACE UMA PONTE ENTRE PRODUTORES E ARTISTAS

LUCAS SOUSA DE MATOS

Prof. Me. Aislan Rafael Rodrigues de Sousa

PICOS, PI
2023

LUCAS SOUSA DE MATOS

MARKETPLACE UMA PONTE ENTRE PRODUTORES E ARTISTAS

Relatório Técnico de Software apresentado como exigência parcial para obtenção do diploma do Curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Piauí, Campus Picos.

Orientador: Prof. Me. Aislan Rafael Rodrigues de Sousa

RESUMO

Este estudo aborda os desafios enfrentados por compositores iniciantes na inserção de suas obras no mercado musical, destacando a dificuldade em atrair o interesse de artistas e empresários. Propõe-se o desenvolvimento de uma plataforma de marketplace para facilitar a conexão entre compositores e potenciais compradores, promovendo oportunidades comerciais e aumentando a visibilidade de músicos independentes. A justificativa para a pesquisa baseia-se nas limitações das grandes gravadoras em reconhecer novos talentos, evidenciando a necessidade de estratégias inovadoras para democratizar o acesso ao mercado musical. A solução proposta visa simplificar a comercialização de composições, tornando a música mais acessível ao consumidor final e contribuindo para uma maior diversidade no cenário musical, ao mesmo tempo que oferece uma alternativa viável ao modelo convencional de distribuição musical.

Palavras-chave: Composição; Dificuldade; Visibilidade ; Mercado.

SUMÁRIO

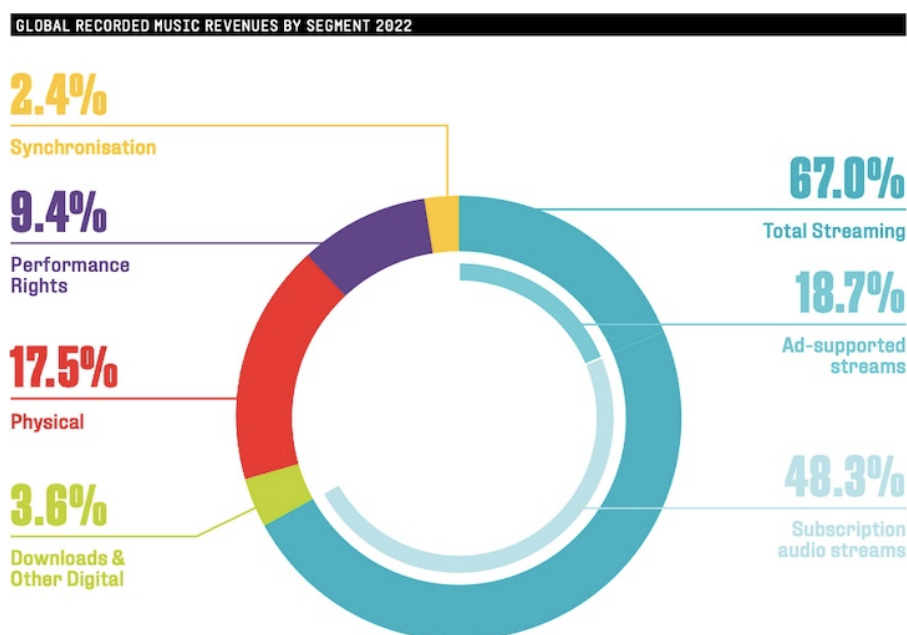
1. INTRODUÇÃO	5
1.1. OBJETIVOS	6
1.1.1. GERAL	6
1.1.2. ESPECÍFICOS	6
2. TECNOLOGIAS ENVOLVIDAS	6
3. MODELAGEM DO PROJETO	8
3.1. LEVANTAMENTO DE REQUISITOS	8
3.1.1 PERSONAS	10
3.1.2 MAPA DE EMPATIA	11
3.1.3 JORNADA DO USUÁRIO	13
3.2. DIAGRAMAS DE CASOS DE USO	14
3.3. DIAGRAMAS DE CLASSE	15
3.4. ARQUITETURA DO SISTEMA	15
3.5. INTERFACE	17
3.5.1 PROTÓTIPOS DE ALTA FIDELIDADE	17
4. SOFTWARE	21
4.1. IMPLANTAÇÃO	21
4.2. TESTES	23
5. CONSIDERAÇÕES FINAIS	25
6. REFERÊNCIA	26

1. INTRODUÇÃO

Este trabalho busca atuar em uma oportunidade de mercado, a justificativa apoia-se na incapacidade das grandes gravadoras observar todos os músicos desconhecidos do grande público, fazendo com que a composição dos músicos seja invisível para sociedade, uma vez que a atenção esteja voltada para artistas de grandes gravadoras já contemplados pela mídia.

O mercado fonográfico está em constante crescimento, e seu principal ponto de arrecadação de direitos autorais hoje é o streaming. De acordo com o *International Federation of the Phonographic industry*(IFPI) e Pro music, o streaming é predominante e representou 67% das receitas totais da indústria musical em todo o mundo, totalizando US\$ 17,5 bilhões como mostra na Figura 1(UBC, 2023). Esse valor de arrecadação se divide basicamente de acordo com os direitos autorais dos Produtores, Associados e Artistas independentes.

Figura 1 - Contribuição de cada segmento musical para as receitas da música global em 2022.



Fonte: IFPI record music

O setor da música atualmente vem se organizando de forma autônoma, se agrupando em associações e redes sociais a fim de buscar um espaço para a produção musical que se encontra fora do mercado comercial, classificando-se como “música independente” (BORBUREMA, 2012).

A solução está inserida no setor da música, um mercado em constante crescimento. Isso evidencia a dificuldade enfrentada pelos compositores para alcançarem as grandes gravadoras e produtores. Através do *marketplace*, pode-se simplificar o processo de visibilidade, tornando mais fácil a música chegar ao consumidor e assim facilitar também a negociação da composição.

1.1. OBJETIVOS

1.1.1. Geral

Desenvolver uma solução tecnológica que funcione como um *marketplace*, conectando compositores e produtores, que possa contribuir para que produtores consigam músicas novas para seus trabalhos, e compositores possam vender suas canções e distribuí-las no mercado.

1.1.2. Específicos

- Investigar sobre o mercado da música;
- Propor uma solução de *marketplace*, onde produtores possam encontrar compositores para comprar os direitos autorais dos trabalhos
- Desenvolver uma solução de *marketplace*, onde produtores possam encontrar compositores para comprar os direitos autorais dos trabalhos;
- Avaliar a solução através de teste de sistema e aceitação.

2. TECNOLOGIAS ENVOLVIDAS

Para criar a interface do sistema, foi utilizado uma combinação de tecnologias, incluindo *HyperText Markup Language*(HTML) e *Cascading Style Sheets*(CSS), além do uso do *framework* BULMA para a estilização do CSS. Além disso, utilizou-se o Vue, um *framework JavaScript* dedicado ao *front end*, para aprimorar a funcionalidade e a interatividade da interface do usuário. As tecnologias estão Ilustradas no Quadro 1.

Quadro 1 - Tecnologias utilizadas no Frontend.

Tecnologias usadas no frontend		
Html	Linguagem de marcação, utilizada para construir sistemas web	
Css	é uma linguagem de marcação que tem como responsabilidade o visual do sistema.	
Bulma	É um framework do css, responsável por contribuir com estilos css já prontos	"^0.9.4"
Vue	É um framework progressivo do javascript, para a construção de interfaces de usuário.	"^3.3.9"
Typescript	O TypeScript é uma linguagem de programação de código aberto da Microsoft, uma extensão do JavaScript que adiciona tipos estáticos opcionais ao código.	"~4.5.5"
Axios	É um cliente HTTP baseado-em-promessas, responsável pela comunicação para o node.js e para o navegador	"^1.6.2"
JWT	É uma forma de autenticação que permite que um servidor verifique a identidade de um usuário sem precisar armazenar informações sobre ele.	"^9.0.2"
Firebase	SDKs multiplataforma de desenvolvimento de apps que ajudam a criar e entregar apps para iOS, Android, Web	"^10.7.2"

Fonte: Próprio Autor.

Para gerenciar as versões do sistema, foi utilizado o GitHub como plataforma de controle de versão, oferecendo uma administração eficaz e colaborativa do código-fonte. No âmbito do back end da aplicação, foi implementado o node.js com Express, além de outras tecnologias que auxiliam na construção do sistema, aplicando os princípios do SOLID para assegurar uma manutenção robusta e escalável ao longo do tempo. Quanto ao banco de dados, foi utilizado o Mysql 2 e para testes de endpoints, será utilizado o Thunder, uma extensão do Visual Studio Code (VSCode), uma plataforma de edição e criação de código-fonte. e suas versões:

Quadro 2 - Tecnologias utilizadas no Backend.

Tecnologias usadas no frontend		
Nodejs	O Node.js é uma plataforma de desenvolvimento de software de código aberto que permite a execução de código JavaScript no lado do servidor.	"^20.10.0"
Express	O Express.js é um framework web para Node.js, projetado para simplificar o processo de criação de aplicativos web e APIs.	"^4.18.2"
Dotenv	O Dotenv é um pacote que serve justamente para gerenciar as variáveis de ambiente dentro de um projeto Node. js	"^16.3.1"
Bcrypt	bcrypt é um método de criptografia do tipo hash para senhas baseado no Blowfish.	"^5.1.1"
jsonwebtoken	JWT é um padrão aberto de mercado definido pela RFC 7519 como um formato compacto e auto suficiente para transmitir informações entre as partes como um objeto JSON	"^9.0.2"
mysql2	O MySQL é um sistema de gerenciamento de banco de dados relacional de código aberto	"^3.7.0"

Fonte: Próprio Autor.

3. MODELAGEM DO PROJETO

Nessa seção serão mostrados os requisitos funcionais e não funcionais do sistema, além de artefatos criados para consolidar os mesmos, e assim tornar uma ideia firme sobre como quais funcionalidades teriam mais prioridades no desenvolvimento do software.

3.1. LEVANTAMENTO DE REQUISITOS

No Levantamento de requisitos foram levados em consideração a pesquisa de campo e a análise do problema abordado entre o compositor e o produtor, assim temos os requisitos funcionais e não funcionais:

- Requisitos Funcionais

1. **Como** usuário, **quero** ter uma forma de me cadastrar de acordo com meu perfil, seja compositor ou artista **para** que eu tenha funcionalidades de acordo com meu perfil.

2. **Como** artista, **quero** cadastrar minha música em uma plataforma que tenha contato com produtores, **para** que ela tenha mais possibilidade de ser ouvida e assim vendida.
3. **Como** compositor, **quero** que antes que eu cadastre a música, tenha um alerta de direitos autorais **para** confirmar que eu realmente esteja sabendo da situação.
4. **Como** produtor, eu **quero** fazer uma busca por gênero, **para** facilitar a busca por meu gosto musical.
5. **Como** artista, eu **quero** ter uma página de perfil **para** que eu possa ter o controle das músicas que já tenho cadastradas.
6. **Como** produtor, eu **preciso** que exista uma página de feeds com todas as músicas, para que eu possa ter acesso às composições e poder escutar cada uma delas.
7. **Como** produtor, **preciso** de uma forma direta, um meio de entrar em contato com o artista **para** negociar a composição.
8. **Como** produtor, **preciso** ter acesso a página de perfil do compositor, **para** que eu possa ouvir outras músicas do mesmo.
9. **Como** artista, **preciso** de uma funcionalidade para editar composição, pois posso ter a necessidade de trocar a capa da música, para passar uma boa impressão.

- **Requisitos Não Funcionais**

1. Boa usabilidade - A interface do usuário deve ser intuitiva e fácil de aprender para usuários novos.
2. Manutenibilidade - Deve ser possível realizar atualizações de software sem interromper o serviço por mais de 30 minutos.
3. Bom desempenho - O sistema deve ter uma boa fluidez entre os componentes, principalmente nas transições entre páginas e na usabilidade de funções.

3.1.1. PERSONAS

As personas são representações fictícias dos clientes do produto, que tem por objetivo entender melhor quem é o usuário da interface, suas características, quais são seus objetivos e as suas necessidades. (Serasaexperian, 2023)

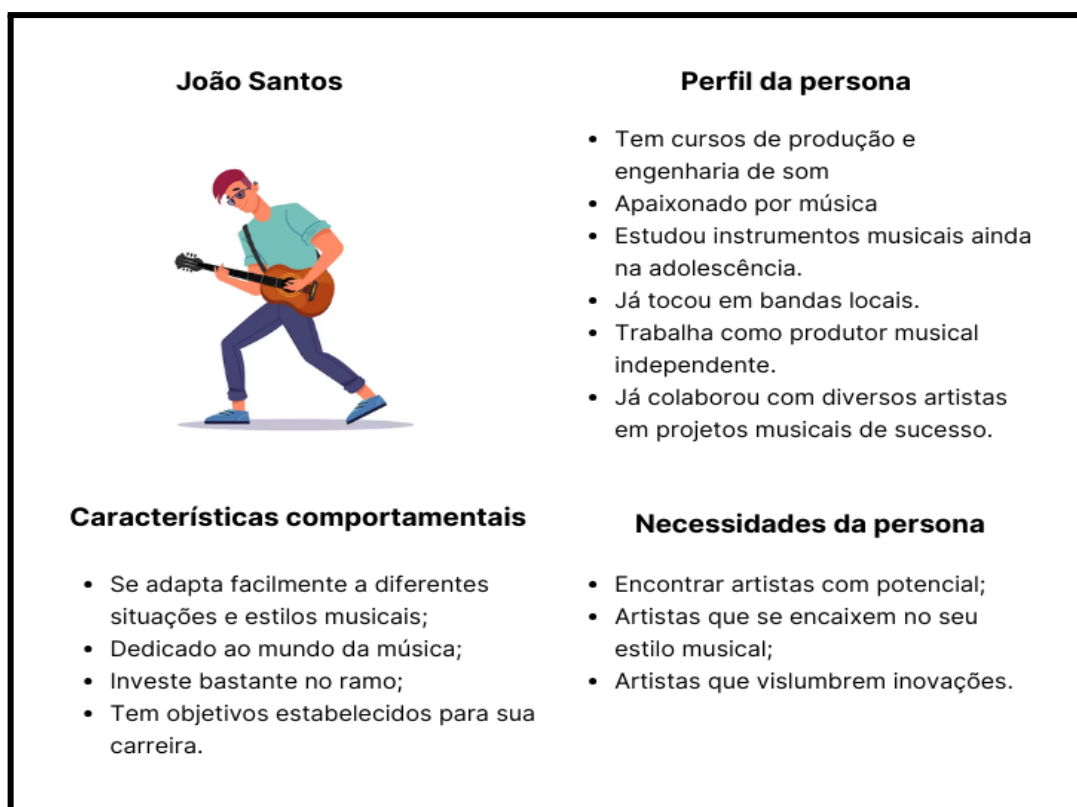
No contexto do sistema proposto, tem-se dois usuários... dois usuários, que possuem problemas diferentes mas que se interligam, um compositor precisa que sua música chegue até um produtor, mas o produtor precisa ter um meio de acesso para que encontre compositores e suas músicas. As Figuras 2 e 3 retratam as duas *personas*.

Figura 2 - Persona do compositor



Fonte: Próprio Autor.

Figura 3 - Persona do produtor



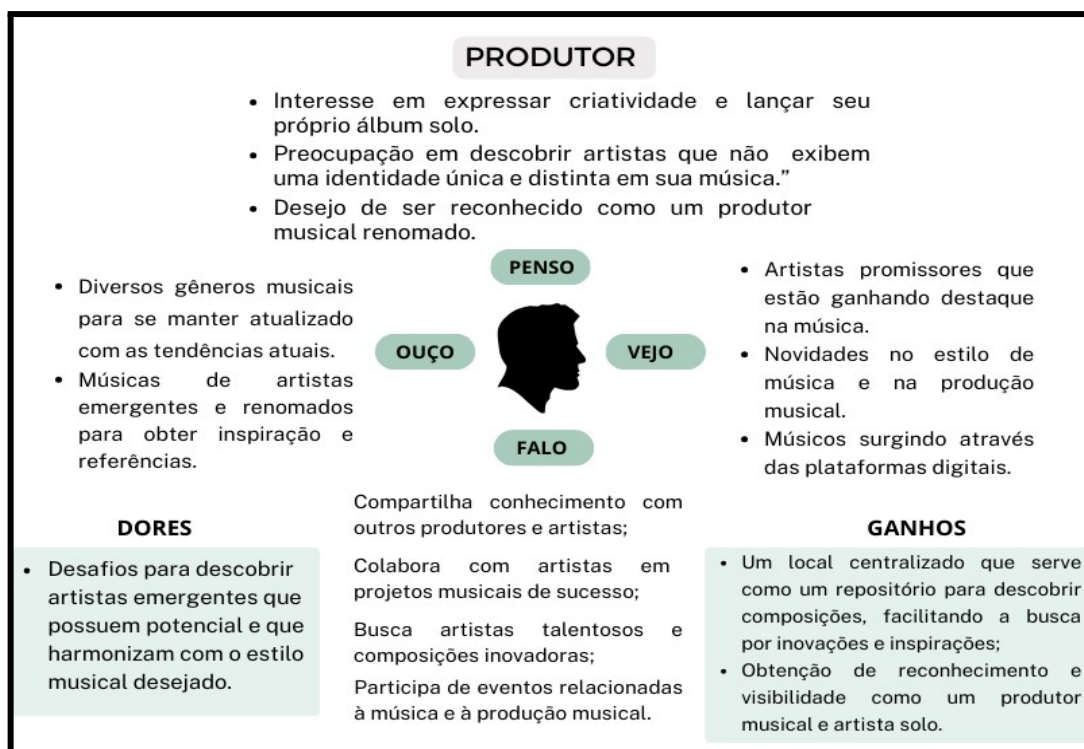
Fonte: Próprio Autor.

3.1.2. MAPA DE EMPATIA

O mapa de empatia é uma valiosa ferramenta visual projetada para proporcionar uma compreensão aprofundada das percepções do usuário. Ele nos guia na exploração do pensamento do usuário, identificando suas preocupações, os obstáculos que enfrenta, as influências do seu ambiente, e as potenciais soluções para resolver seus problemas. (Lima, 2021)

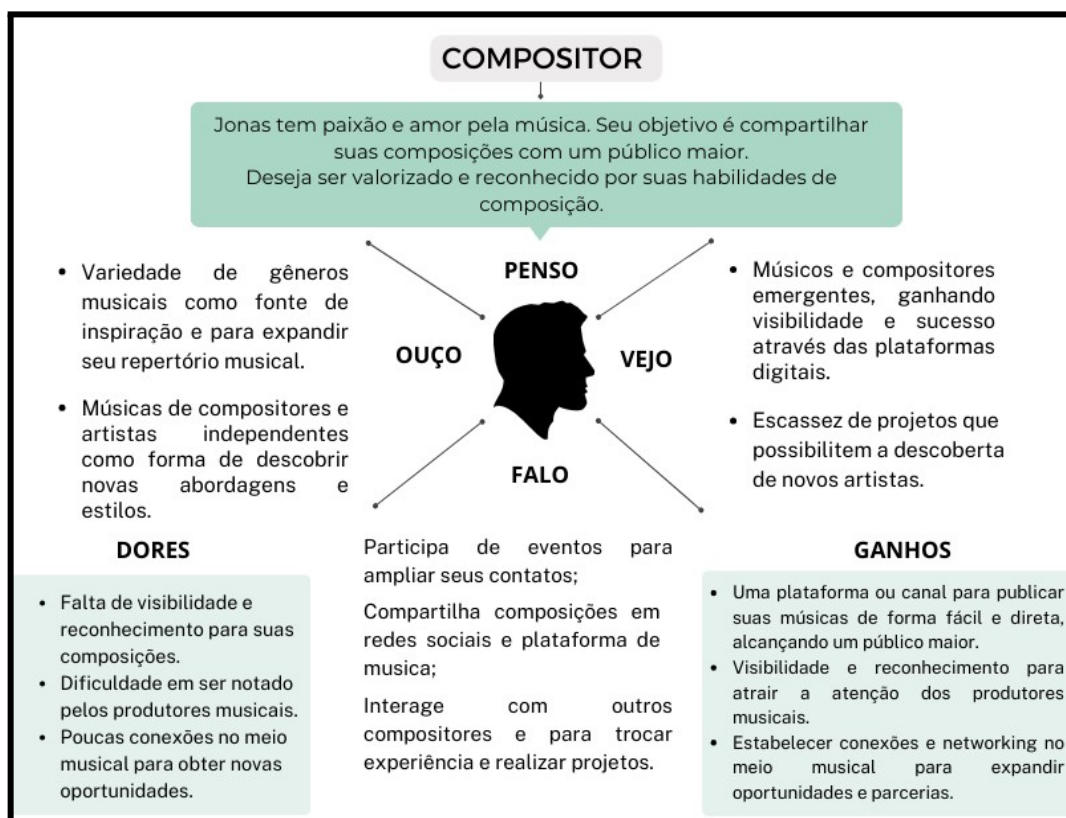
Este mapa de empatia é organizado em seis categorias fundamentais, abrangendo o que o usuário percebe visualmente, auditivamente, verbaliza e prática, quais são suas dores emocionais e as recompensas que enxerga ao considerar uma solução para suas necessidades. Abaixo, apresentamos nas Figuras 4 e 5 os mapas de empatia elaborados para os respectivos usuários:

Figura 4 - Mapa de empatia do produtor



Fonte: Próprio Autor.

Figura 5 - Mapa de empatia do compositor



Fonte: Próprio Autor.

3.1.3. JORNADA DO USUÁRIO

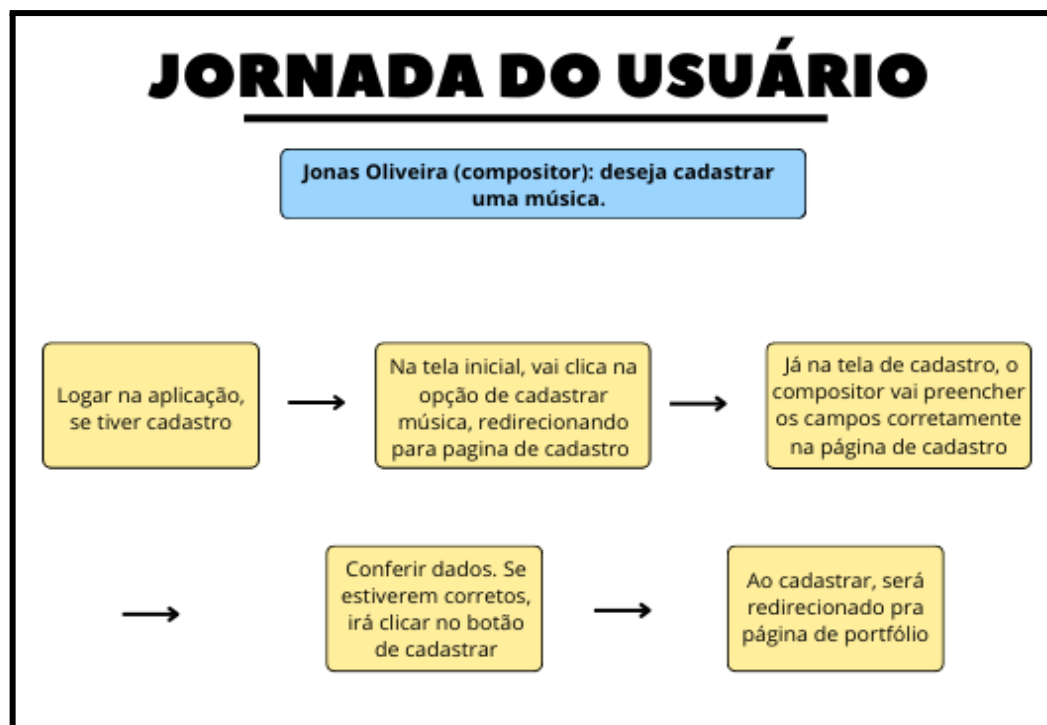
Nesta seção, abordaremos o percurso que um usuário específico seguirá para alcançar seu objetivo final. O fluxo da jornada do usuário é, essencialmente, uma representação visual passo a passo das interações do usuário dentro do sistema, visando executar uma determinada funcionalidade essencial. As Figuras [6, 7] ilustram o fluxo das personas mencionadas nos tópicos anteriores, juntamente com suas decisões ao longo do caminho

Figura 6 - Jornada do produtor



Fonte: Próprio Autor

Figura 7 - Jornada do compositor

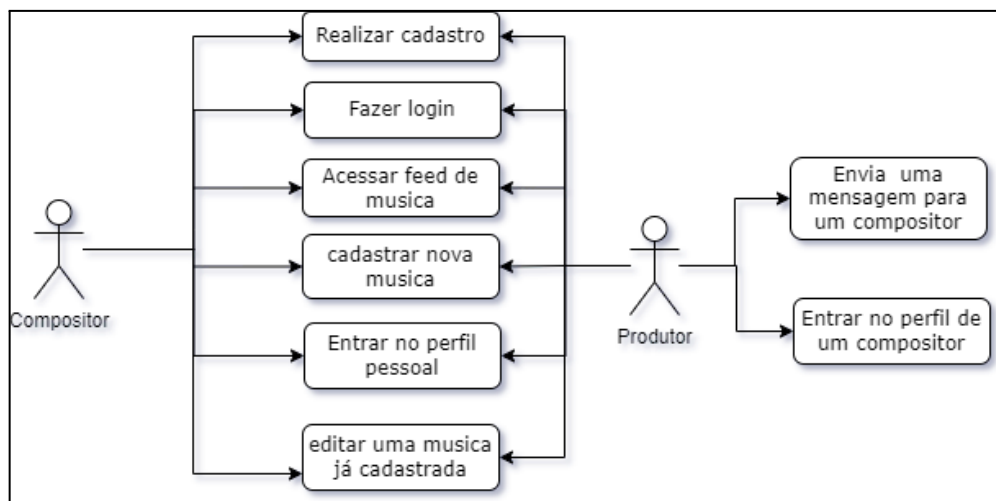


Fonte: Próprio Autor

3.2. DIAGRAMAS DE CASOS DE USO

O diagrama de caso de uso é uma representação específica de quais serão as funcionalidades do sistema, quem serão os atores (Usuários), e como eles irão se relacionar com essas atividades. A Figura 8 apresenta esse diagrama, ilustrando as funcionalidades que o compositor, e o produtor poderá acessar.

Figura 8 - Diagrama de caso de uso.



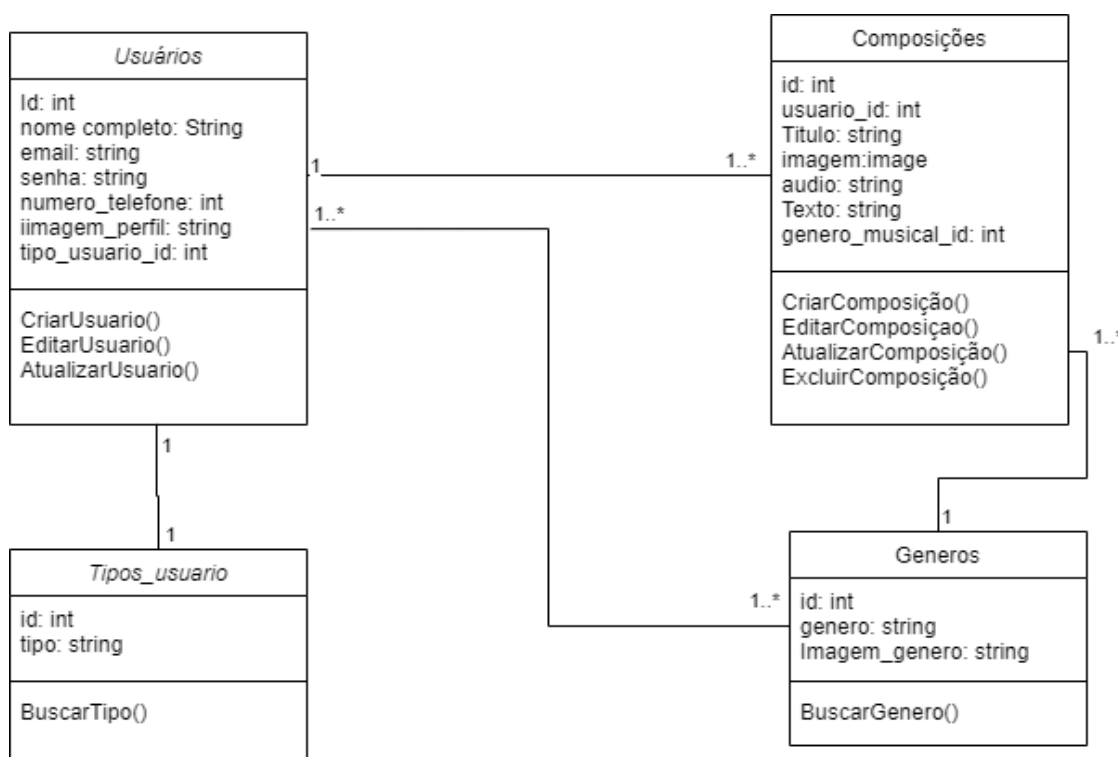
Fonte: Próprio Autor.

3.3. DIAGRAMAS DE CLASSE

O diagrama de classes traz uma modelagem estática dos objetos e modelam a estrutura do sistema de forma simples, através dele pode-se tornar mais clara a visão de como os as funcionalidades vão interagir entre si por meio do relacionamento no diagrama de classe.

Cada classe representa um objeto no sistema, os atributos descrevem características da classe e por fim os métodos, definem o comportamento de cada classe. A Figura 9 ilustra o diagrama de classe deste projeto.

Figura 9 - Diagrama de classes



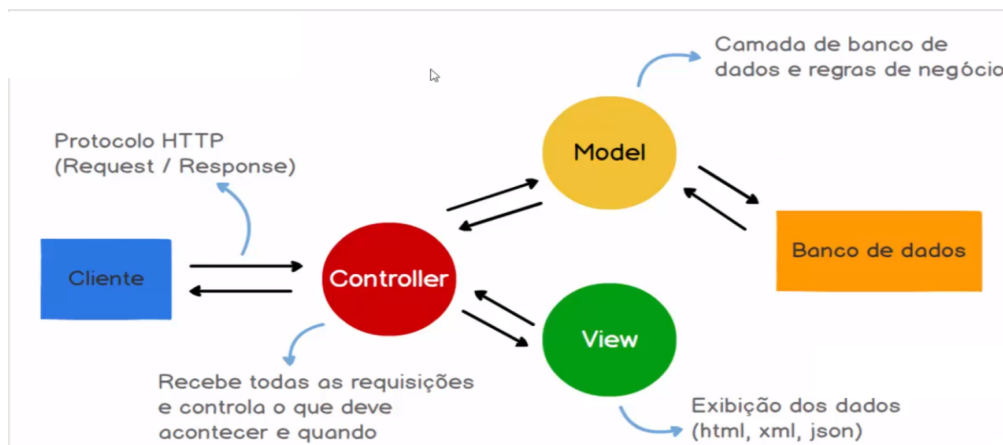
Fonte: Próprio Autor.

3.4. ARQUITETURA DO SISTEMA

No desenvolvimento deste projeto, a estrutura do *back end* foi utilizado o modelo *Model View Controller* (MVC), que divide a aplicação em três componentes distintos. O Modelo, responsável pela manipulação dos dados e lógica de negócios, além de interagir com o banco de dados para gerenciar informações. A Visão, por sua vez, cuida dos dados que vão ser mostrados para os usuários, por meio de componentes *Vue.js*. O Controlador atua como intermediário, respondendo a

eventos do usuário e coordenando a comunicação entre o Modelo e a Visão. A Figura 10 explica de forma resumida como funciona este modelo.

Figura 10 - Modelo mvc

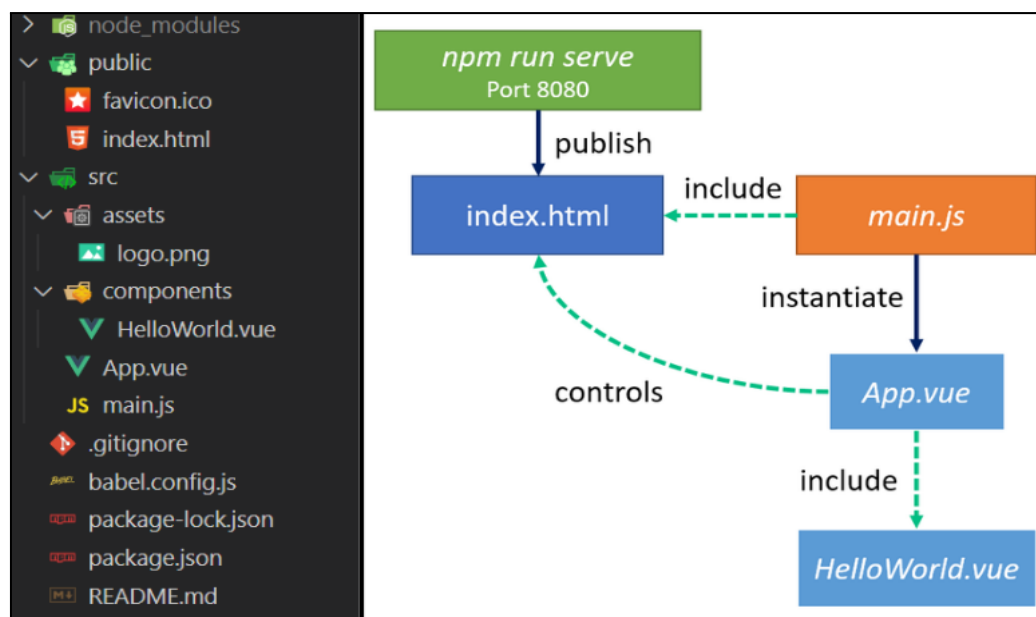


Fonte: Próprio Autor.

A implementação do back end foi fundamentada na adoção de uma REST API (Interface de Programação de Aplicações baseada em Transferência de Estado Representacional). Esta abordagem facilita a comunicação eficiente entre o *front end* e o *back end*, estruturando cada entidade como um recurso e mapeando operações CRUD para métodos HTTP correspondentes. Além disso, medidas robustas de segurança, como autenticação baseada em tokens JWT e controle de acesso por papéis, foram feitas para garantir a integridade e proteção dos dados.

No frontend, foi utilizada a estrutura padrão de pastas do Vue.js conforme apresentado na Figura 11.. A construção modular da aplicação, baseada em componentes reutilizáveis, proporciona flexibilidade e facilidade de manutenção. O roteamento dinâmico simplifica a navegação entre diferentes seções da aplicação, promovendo uma transição suave e eficiente. Em conjunto, o modelo MVC, a REST API e o Vue.js estabelecem uma arquitetura escalável, que torna fácil a manutenção do projeto e futuras expansões.

Figura 11 - organização de pastas e fluxo de execução



Fonte: Próprio Autor.

Os códigos do front end e back end estão armazenados no Github separadamente, disponíveis em:

- tentilhao-music: [mattos251/tentilhao-music \(github.com\)](https://github.com/mattos251/tentilhao-music)
- tentilhao_backend: <https://github.com/mattos251/tentilhao-backend>

3.5. INTERFACE

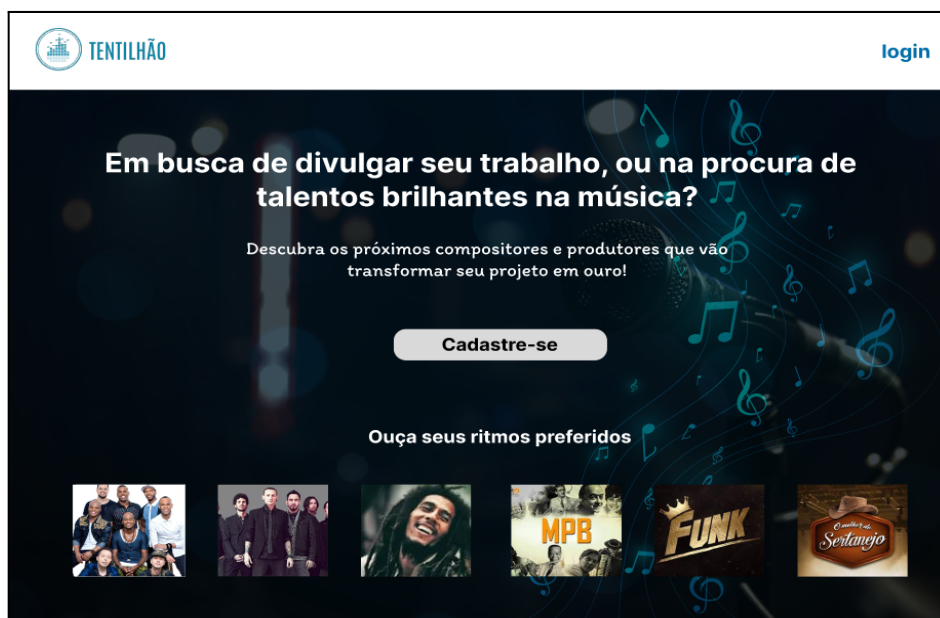
Toda a interface do sistema foi construída com as tecnologias Html, Css, Bulma e Vuejs 3, pensando na fluidez e suporte da plataforma, além dessas tecnologias possuírem boas documentações para fins de solução de qualquer problema que poderia acontecer.

3.5.1. PROTÓTIPOS DE ALTA FIDELIDADE

Um protótipo de alta fidelidade é uma versão visual e interativa de um produto ou sistema, criada para se parecer muito com a versão final em termos de design, disposição e funcionalidades. Dessa forma pode haver mudança no design ainda no desenvolvimento, mas o estilo e palhetas de cores seguem formando padrões. A seguir temos as telas principais do sistema, seguindo os fluxos principais tanto do perfil do compositor como do produtor.

- **Compositor/ Produtor:** Na tela principal, o usuário poderá se cadastrar, clicando em “cadastrar”, ou fazer “Login”, caso tenha conta, conforme a Figura 12.

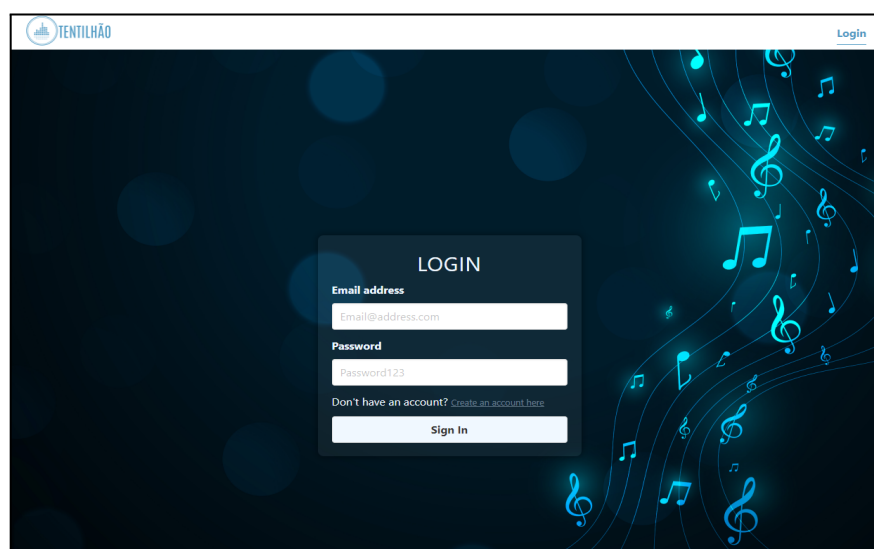
Figura 12 - Landing page



Fonte: Próprio Autor

- **Compositor/ Produtor:** Página para o usuário realizar o login e ter acesso a página inicial, ilustrada na Figura 13.

Figura 13 - Página de Login



Fonte: Próprio Autor

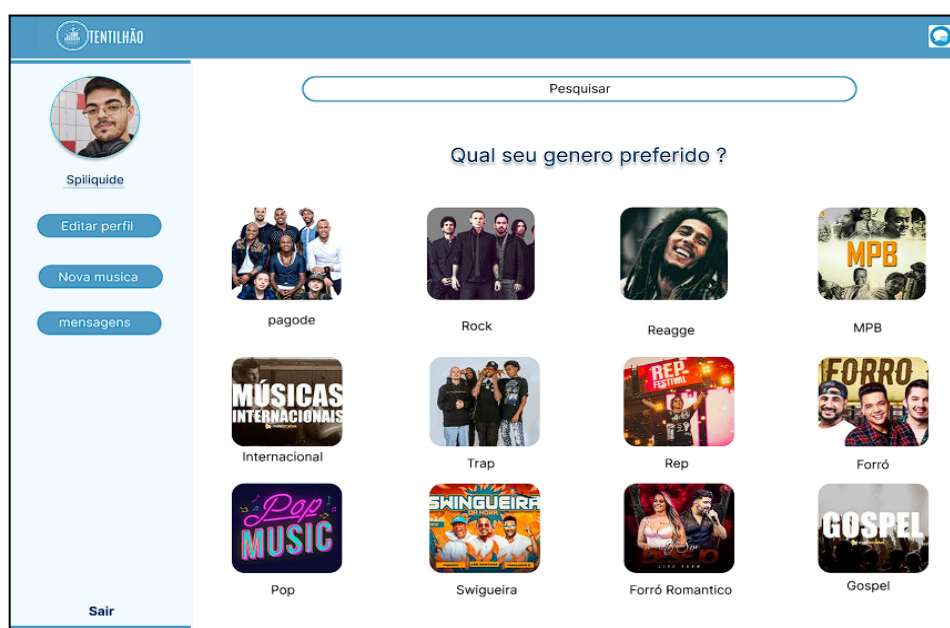
- **Compositor/Produtor:** Caso não tenha uma conta, ele vai à Página de Cadastro de Usuário e realiza o registro, conforme a Figura 14.

Figura 14 - Cadastro de usuário

Fonte: Próprio Autor

- **Compositor/Produtor:** Página de escolha de gêneros ao realizar o login, de acordo com a Figura 15.

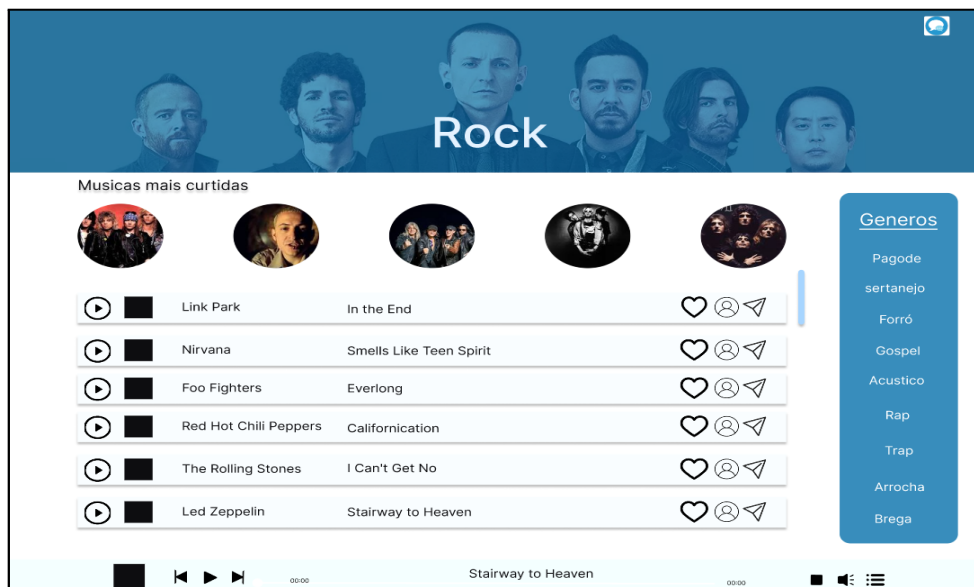
Figura 15 - Página de feeds



Fonte: Próprio Autor

- **Compositor/Produtor** Lista de composições com base no gênero escolhido, ilustrado na Figura 16.

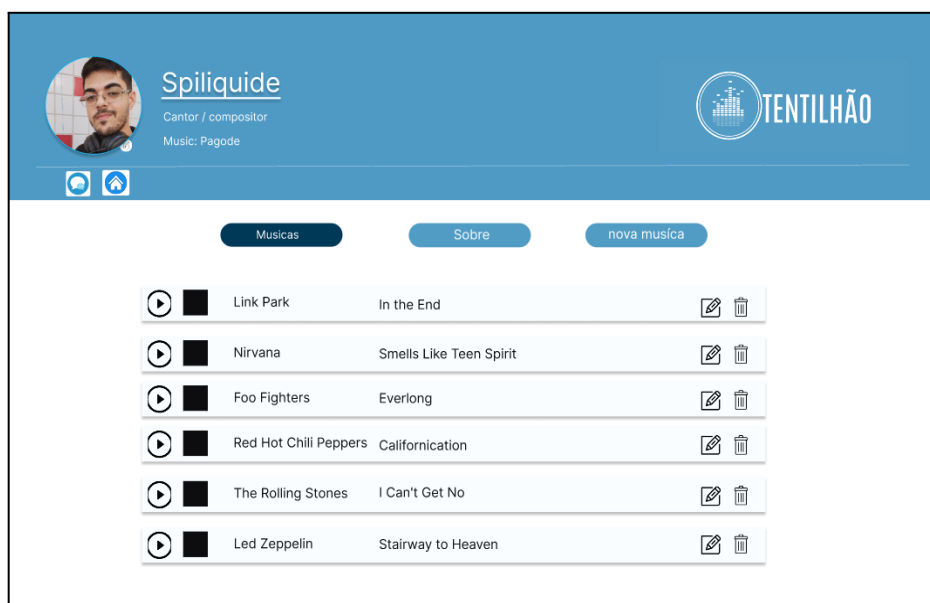
Figura 16 - Página de listagem de músicas



Fonte: Próprio Autor

- **Compositor/ Produtor** Na Figura 17, está a Página do perfil, o usuário poderá escutar suas composições, editar, excluir ou cadastrar uma nova.

Figura 17 - Página de listagem de músicas do feed.



Fonte: Próprio Autor

- **Compositor/ Produtor** Página de cadastro de composições, ilustrada na Figura 18.

Figura 18 - Página de cadastro.

Fonte: Próprio Autor

4. SOFTWARE

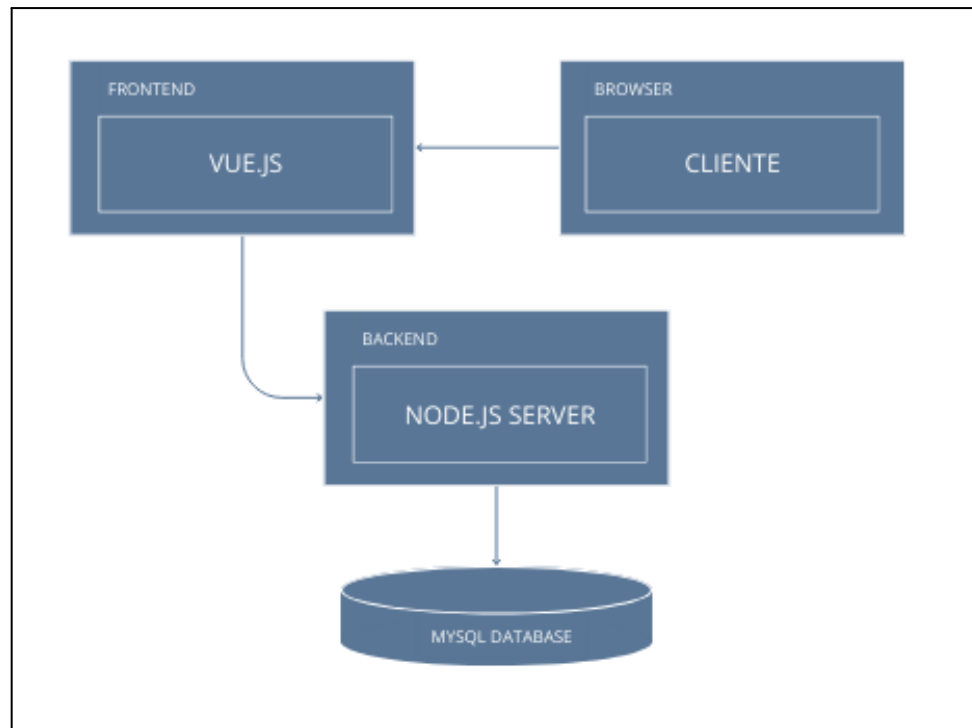
Neste tópico será apresentado o modelo de implantação do software, quais tecnologias foram usadas no desenvolvimento do software e os testes aplicados para consolidar o seu funcionamento como todo.

4.1. IMPLANTAÇÃO

Diagramas de implantação, pertencem ao UML, e tem o papel de descrever a arquitetura de execução de um sistema, evidenciando elementos cruciais, como nós de hardware e software, além do *middleware* que facilita suas conexões. Esses diagramas são valiosos para a visualização da infraestrutura física de um sistema, permitindo uma compreensão abrangente de sua implantação no hardware específico.

No processo de implantação deste sistema, o desenvolvimento do frontend foi iniciado primeiro. Cada tela criada passava por uma avaliação e validação para consolidar o processo. A Figura 19 ilustra cada etapa e a comunicação entre as partes do sistema.

Figura 19 - Diagrama de implementação



Fonte: Próprio Autor.

- Browser
 - Representa o cliente ou usuário que interage com a aplicação através do navegador.
- Front end (Vue.js)
 - O código Vue.js é executado no navegador do cliente.
 - Interage com o backend por meio de solicitações HTTP.
- Back end(Node.js)
 - O código Node.js é executado no servidor.
 - Utiliza o framework Express para gerenciar as rotas e as solicitações HTTP.
 - Conecta-se ao banco de dados para recuperar ou armazenar dados.
- Servidor de Banco de Dados(Mysql 2)
 - Armazena e gerencia os dados do aplicativo, conectando com o banco de dados Mysql 2.

4.2. TESTES

Para a implementação e certificação de funcionalidades, foi aplicado o método de teste de *Scale Usability Scale(SUS)*, criado por John Brooke, em 1986, que traz uma ideia de pontuação de usabilidade em escala por meio de um questionário. Esse questionário é composto por 10 questões, distribuídas para um grupo de pessoas, contendo perguntas positivas e negativas.

Para chegar a um resultado, cria-se um agrupamento de questões ímpares e pares, sendo que as respostas das questões ímpares são subtraídas por -1, e os resultados obtidos pelas questões pares, são feitas como “5 - Resposta da questão”. Com a soma dos resultados obtidos é feita uma multiplicação, “Resultado x 2.5”. Os cálculos realizados tem por objetivo medir esses critérios:

- Facilidade de aprendizagem;
- Eficiência;
- Facilidade de memorização;
- Minimização dos erros;
- Satisfação;

Entendendo o método de maneira mais profunda, foi aplicado o questionário com as seguintes afirmações e resultados:

1. O sistema possui boa fluidez, entre as páginas!
2. Eu achei o sistema fácil de usar!
3. O sistema apresenta erros críticos que influenciam na usabilidade!
4. O sistema possui todas funcionalidades apresentadas na descrição!
5. Eu como compositor, usaria esse sistema!
6. O sistema possui todas as funcionalidades necessárias para atender as necessidades de um produtor e compositor!
7. Eu indicaria esse sistema para um amigo compositor/produtor!
8. Eu imagino que as pessoas aprenderão como usar esse sistema rapidamente!
9. O sistema está funcionando muito bem!
10. Eu acho o sistema desnecessariamente complexo!

O questionário mencionado foi aplicado a um total de 16 participantes, predominantemente constituído por profissionais do setor musical, incluindo cantores e compositores, assim como por indivíduos que, embora não atuem no domínio da

música, têm o hábito de ouvir canções diariamente. Após realização dos cálculos, obteve-se os resultados apresentados na Tabela 3..

Quadro 3 - Tabela de resultado SUS

SUS FINAL SCORE			
Participante 1	80	Participante 9	100
Participante 2	100	Participante 10	100
Participante 3	95	Participante 11	65
Participante 4	77,5	Participante 12	72,5
Participante 5	97,5	Participante 13	100
Participante 6	90	Participante 14	80
Participante 7	100	Participante 15	82,5
Participante 8	100	Participante 16	80
Média total: 88,75			

Fonte:Próprio Autor

O resultado geral obtido teve uma média de 88,75 na escala do SUS de 0 a 100, indicando um bom desempenho do sistema criado e seu potencial como produto. No formulário também foi sugerido melhorias, como:

- Mais verificações no onboarding;
- Otimize o site para dispositivos móveis;
- Imagem do perfil está um pouco distorcida;
- O cadastro de música está um pouco demorado;
- Cores muito claras;
- mudar placeholder da descrição do perfil;
- mudar botão "cadastrar" para "atualizar";

Dessas sugestões algumas já foram implementadas, e outras serão no decorrer do tempo.

5. CONSIDERAÇÕES FINAIS

Nesse documento foi apresentado artefatos sobre a pesquisa de mercado, mostrando como funciona o mercado fonográfico, quais são seus pontos fortes e fracos hoje, a partir da identificação das áreas mais carentes, surgiu o tema central, que serviu como ponto de partida para a exploração abrangente do problema e, conseqüentemente, para o desenvolvimento de uma solução viável. No processo, de entender como seria o desenvolvimento do software, foi criado artefatos como personas, história de usuário, mapa de empatia, jornada de usuário, captação de requisitos, produto de backlog, documentos para ter uma ideia de quais funcionalidades seriam necessárias para suprir a necessidade dos usuários.

Após a maturação da ideia central, estruturou-se o marketplace, definindo as tecnologias que poderiam ser usadas no *front end*, quais poderiam ser usadas no *back end*, qual banco de dados seria melhor para aplicação, relacional, ou um não relacional. Decisões que foram tomadas a partir de mais pesquisas e assim criados artefatos como, modelo de classes, diagrama de implementação e protótipos de alta fidelidade que proporcionam uma ótima ideia do produto.

Por fim, foram conduzidos testes de usabilidade envolvendo usuários reais do sistema, aplicando o método SUS. Cada etapa mencionada foi implementada e, em conjunto, resultou em uma média de 88,75 numa escala de 0 a 100. Esses resultados confirmam o potencial do sistema para ingressar no mercado, destacando áreas passíveis de aprimoramento. Algumas melhorias já estão em execução, enquanto outras serão implementadas ao longo do tempo, com base em testes adicionais.

6. REFERÊNCIAS

CUNHA, Glória; MARTINS, Maria Cecília. Tecnologia, Produção & Educação Musical: Descompassos e Desafinos. In: Anais do Congresso RIBIE n. IV. 1998.

Federação Internacional da Indústria Fonográfica (ifpi). (2022, May 18). global musicreport.2022.https://www.ifpi.org/wp-content/uploads/2022/04/IFPI_Global_Music_Report_2022-State_of_the_Industry.pdf

GANDRA, Alana, Relatório do Ecad mostra efeito da pandemia no mercado de shows, Agência Brasil, 2021, Disponível em:<https://agenciabrasil.ebc.com.br/geral/noticia/2021-09/relatorio-do-ecad-mostra-efeito-da-pandemia-no-mercado-de-shows>. Acesso em: 30/06/2022

GARLAN, David; PERRY, Dewayne E. Introduction to the special issue on software architecture. **IEEE Trans. Software Eng.**, v. 21, n. 4, p. 269-274, 1995.

JÚNIOR, Francisco Robério Fernandes. Direitos autorais nas obras musicais sob a ótica da Lei nº 9.610 de 1998. Revista Acadêmica Escola Superior do Ministério Público do Ceará, v. 11, n. 2, p. 129-146, 2019.

KAZMAN, Rick; ASUNDI, Jai; KLEIN, Mark. Quantifying the costs and benefits of architectural decisions. In: Proceedings of the 23rd International Conference on Software Engineering. ICSE 2001. IEEE, 2001. p. 297-306.

LIMA, Gabriela. Guia prático de como montar um Mapa de Empatia, Alura, 2021, Disponível em:<<https://www.alura.com.br/artigos/guia-pratico-como-montar-mapa-empatia>>. acesso em 05 de janeiro de 2024.

MOSCHETTA, Pedro Henrique; VIEIRA, Jorge. Música na era do streaming: curadoria e descoberta musical no Spotify. Sociologias, v. 20, p. 258-292, 2018.

Serasa Experian. “Personas: entenda o que são e como aplicar no seu negócio.”

Serasa Experian, 2021. Disponível em:

<<https://www.serasaexperian.com.br/conteudos/marketing/persona/>> . Acesso em:

10 de janeiro 2024.

SILVA, Janaína Tenório Trancoso Tavares. *Marketplace*: a grande salva-vidas das lojas virtuais.

SOLER, Alessandro, MÚSICA GRAVADA GERA US\$ 26 BI NO MUNDO EM 2022; BRASIL VOLTA A TOP 10, UBC, Disponível em:

<<https://www.ubc.org.br/publicacoes/noticia/21434/musica-gravada-gera-us-26-bi-no-mundo-em-2022-brasil-volta-a-top-10>> , Acesso em: 17 de fevereiro de 2024.

Receita global de música gravada cresce 7,4%.ifpi, 2021. Disponível em:

<<https://www.ifpi.org/ifpi-issues-annual-global-music-report-2021> />. Acesso em: 11 de maio. de 2022.

Relatório Anual 2021, 2021. Disponível em:

<<https://www4.ecad.org.br/wp-content/uploads/2022/06/RelatorioAnual2021.pdf> />. Acesso em: 05 de julho de 2022.