

**UNIPÊ**

**Sistemas Cognitivos**

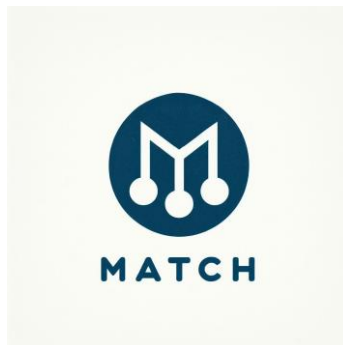
Raquel Vieira Campos

João Victor de Luna

Livia Lima

Ana Rachel

João Lucas da Silva



João Pessoa, 22 de novembro de 2023.

### Equipe de desenvolvimento

Nome	Função
João Victor de Luna	Desenvolvedor front-end
Livia Lima	Desenvolvedor back-end
Ana Rachel	Desenvolvedor back-end
João Lucas	Desenvolvedor back-end
Raquel Campos	Documentação e protótipo de tela

### Cronograma do projeto

Data	Descrição
25/10/2023	Definição da equipe
01/11/2023	Definição do tema
21/11/2023	Testes da versão final.
22/11/2023	Entrega e apresentação.

## 1. Introdução

Este documento especifica os requisitos do sistema MATCH. Fornecendo as informações necessárias para o projeto e implementação, assim como para a realização dos testes e homologação do sistema.

- O Projeto consiste na criação de um programa voltada à identificação de músicas de nome e autoria desconhecida. A aplicação deverá fornecer ao usuário o nome da música que ele deseja descobrir, nome do cantor e a transcrição do áudio.

Link do repositório: [github.com/joaovictorlfs/MatchAudioRecognition](https://github.com/joaovictorlfs/MatchAudioRecognition)

## 2. Descrição geral da Aplicação

Quando o usuário ouve uma música e deseja saber o nome e o cantor, ou até mesmo quando a música está em outro idioma e ele gostaria de saber a letra, ele pode utilizar o MATCH para identificar essas informações. O algoritmo de reconhecimento de músicas recebe auxílio da lib shazamio para identificar a música através do seu ID e também o modelo Whisper para fazer a transcrição do áudio.

### 2.1. Gravação e Envio da Amostra de Áudio:

- O usuário clica no botão do Match e o coloca próximo à fonte de áudio, para gravar o que está ouvindo.
- O aplicativo grava uma breve amostra de 10 segundos do áudio.
- O áudio é enviado em formato de bytes para a lib do shazamio, que irá identificar o ID daquela música.

### 2.2. Identificação e Retorno de Informações:

- Quando o ID é encontrado, ou seja, quando ocorre um Match, a aplicação retorna as informações da música que estão associadas ao ID, como título, artista, e imagem. Já o whisper retorna a transcrição do áudio.

## 3. Descrição de dados e tecnologias

### 3.1 Dados de entrada

Descrição: O dado de entrada é a gravação feita pelo usuário.

Atores: Usuário

Entradas e pré-condições: Estar conectado à internet e ter microfone.

Atributo	Tipo
Áudio	Wav

### 3.2 Dados de saída

Descrição: Nome da música, cantor(a), imagem, transcrição

Atores: Shazamio

Entradas e pré-condições: O ID da música precisa ser reconhecido.

Saídas e pós-condição: Dados associados ao ID encontrado.

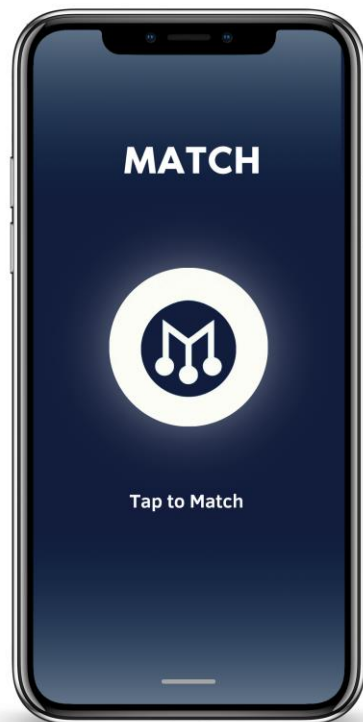
Atributo	Tipo
Nome	string
Cantor(a)	string
Transcrição	string
Imagem	jpg

### 3.3 Tecnologias

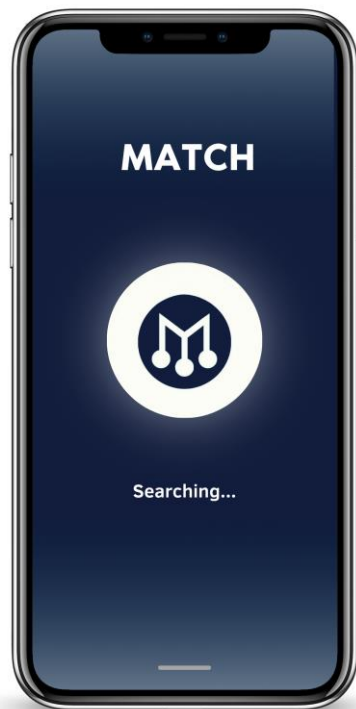
- **Python:** o código da aplicação foi escrito em Python, utilizando a biblioteca Shazamio.
- **Tkinter:** A interface gráfica do projeto foi feita utilizando a biblioteca Tkinter.
- **Whisper:** a transcrição é feita pelo modelo pré-treinado Whisper (small).

## 4. Prototipações

### 4.1.



4.2



4.3.



4.4.

