



Fowyv, Find Others With Your Voice

Bernardo Poiares

Orientadora Paula Graça

Relatório final realizado no âmbito de Projeto e Seminário,
do curso de licenciatura em Engenharia Informática e de Computadores
Semestre de Verão 2019/2020

Setembro de 2020

Instituto Superior de Engenharia de Lisboa

Licenciatura em Engenharia Informática e de Computadores

Fowyv, Find Others With Your Voice

39100 Bernardo António Silva Poiares

Orientadora: Paula Graça, ISEL

Relatório final realizado no âmbito de Projeto e Seminário,
do curso de licenciatura em Engenharia Informática e de Computadores
Semestre de Verão 2019/2020

Setembro de 2020

Resumo

O projeto Fowyv trata-se de um sistema que permite aos seus utilizadores conhecerem virtualmente novas pessoas, garantindo uma experiência mais pessoal e íntima em comparação com outras opções existentes atualmente. Sendo que, Fowyv traz para o mundo virtual o foco num meio fulcral de interação e comunicação entre pessoas, rege-se na característica humana mais íntima: a voz.

Ou seja, o projeto baseia-se em dar ênfase à voz, tentando passar ao utilizador uma experiência mais real como se tratasse de uma conversa entre duas pessoas que se encontrariam num café pela primeira vez, oferecendo as ferramentas e o local que permitam a relação evoluir mesmo existindo a distância física.

Fowyv é composto por quatro componentes: uma aplicação móvel que serve de interface funcional para os utilizadores, uma aplicação servidora responsável por disponibilizar os serviços que permitam o acesso e o armazenamento dos dados à aplicação móvel, uma base de dados para o armazenamento de dados e um serviço de armazenamento para o armazenamento de ficheiros de áudios.

Na elaboração deste projeto deparei-me com diferentes dificuldades e barreiras. Tive sobretudo limitações tecnológicas por trabalhar com recursos que me eram desconhecidos até então, acabando este por ser um ponto positivo visto que alarguei o meu conhecimento e sendo eles recursos que poderei voltar a utilizar futuramente.

O projeto foi finalizado cumprindo os objetivos traçados, contudo existe bastante espaço de melhorias. Além disso, poderiam ser introduzidas novas funcionalidades que tornassem a aplicação mais apelativa e mais rica de forma a aumentar a sua procura e a ser algo de uso contínuo/constante e não uma aplicação igual a tantas outras esquecida no telemóvel do seu utilizador.

Palavras-chave: relacionamento *online*, voz.

Abstract

The Fowyv project is a system that allows users to meet virtually new people, ensuring a more personal and intimate experience compared to other options currently available. Fowyv brings to the virtual world a crucial way of interaction and communication between new people, it brings the most intimate human characteristic: the voice.

The project is based on the voice, trying to give to the user a more real experience as a conversation between two people who would meet in a bar for the first time, providing the tools and the place that allows the relationship to evolve even with the physical distance.

Fowyv consists of four components: a mobile application that serves as a functional interface for users, a server application responsible for providing services that allow access and storage of data to the mobile application, a database for data storage and a storage service for storing audio files.

In preparing this project I came across with different difficulties and barriers. I mainly had technological limitations for working with resources that were unknown to me until then, turning out to be a positive point since I broadened my knowledge and since they are resources that I will be able to use in the future.

The project was completed in compliance with the objectives set, however there is plenty of room for improvement. In addition, new features could be introduced to make the application more appealing and richer in order to increase its demand and to be something of constant use and not an application like many others forgotten on its user's mobile phone

Keywords: online relationship; voice.

Agradecimentos

Tenho a agradecer sobretudo à minha Orientadora Paula Graça pelo seu apoio e suporte, ambos indispensáveis na elaboração e concretização deste projeto.

Devo agradecer à minha irmã, Margarida Poiares, por me ter apoiado com a parte estética e gráfica do projeto.

E também não posso deixar de agradecer à minha família que sempre me motivou e incentivou a terminar este capítulo da minha vida académica.

Índice

RESUMO	V
ABSTRACT	VII
AGRADECIMENTOS	IX
ÍNDICE.....	XI
LISTA DE FIGURAS	XIII
1. INTRODUÇÃO	1
1.1 ENQUADRAMENTO	1
1.2 SITUAÇÃO ATUAL	1
1.3 OBJETIVOS	2
1.4 ORGANIZAÇÃO DO DOCUMENTO	2
2. FORMULAÇÃO DO PROBLEMA	3
2.1 TERMINOLOGIA	3
2.2 REQUISITOS	3
2.2.1 Comunicação tempo real	3
2.2.2 Simplicidade e Intuitiva.....	3
2.2.3 Armazenamento de dados	3
2.2.4 Versatilidade	4
2.2.5 Escalabilidade.....	4
2.3 DESCRIÇÃO DA APLICAÇÃO	4
2.3.1 Autenticação do utilizador	5
2.3.2 Verificar áudios de outros utilizadores.....	5
2.3.3 Interagir com correspondências	6
2.3.4 Configurações.....	7
2.4 TECNOLOGIAS UTILIZADAS	8
2.4.1 Aplicação móvel	8
2.4.2 Aplicação servidora	9
2.4.3 Base de dados	9
2.4.4 Serviço de armazenamento.....	9
3. SOLUÇÃO PROPOSTA - GRANDES IDEIAS	10
3.1 ARQUITETURA DO SISTEMA	10
3.2 DESCRIÇÃO DETALHADA DA SOLUÇÃO	11
3.2.1 Aplicação móvel	13

3.2.2	<i>Aplicação Servidora</i>	16
3.2.3	<i>Base de dados</i>	18
3.2.4	<i>Serviço de alojamento de ficheiros</i>	20
3.3	DISPONIBILIZAÇÃO DAS APLICAÇÕES E SERVIÇOS	20
4.	AVALIAÇÃO EXPERIMENTAL	21
4.1	TESTES FUNCIONAIS	21
4.2	ANÁLISE DOS RESULTADOS OBTIDOS	38
5.	CONCLUSÕES	41
	REFERÊNCIAS	43
A.1	WIREFRAMES	44

Lista de Figuras

<i>Figura 1 - Casos de Uso considerados para a experiência dos utilizadores</i>	<i>4</i>
<i>Figura 2 – Use case Login</i>	<i>5</i>
<i>Figura 3 – Use case Verificar áudios de outros utilizadores</i>	<i>6</i>
<i>Figura 4 – Use case Interagir com correspondências</i>	<i>7</i>
<i>Figura 5 – Use case Configurações.....</i>	<i>8</i>
<i>Figura 6 - Arquitetura do sistema proposto</i>	<i>10</i>
<i>Figura 7– Esquema sequencial da comunicação entre a aplicação móvel e a aplicação servidora</i>	<i>11</i>
<i>Figura 8 – Esquema sequencial de comunicação através de websockets entre a aplicação móvel e a aplicação servidora</i>	<i>12</i>
<i>Figura 9 – Esquema sequencial do armazenamento de uma mensagem de áudio</i>	<i>13</i>
<i>Figura 10 -Estrutura da aplicação móvel</i>	<i>14</i>
<i>Figura 11 - Estrutura da aplicação servidora para a exposição da API HTTP</i>	<i>16</i>
<i>Figura 12 - Estrutura da aplicação servidora para a exposição dos websockets</i>	<i>17</i>
<i>Figura 13 - Estrutura dos documentos das coleções</i>	<i>19</i>
<i>Figura 14 – Ecrã inicial da aplicação</i>	<i>21</i>
<i>Figura 15 – Ecrã de autenticação do utilizador.....</i>	<i>22</i>
<i>Figura 16 – Ecrã de criação de conta do utilizador.....</i>	<i>23</i>
<i>Figura 17 – Ecrã de definição do perfil do utilizador</i>	<i>24</i>
<i>Figura 18 – Ecrã com elemento modal para gravação de áudio</i>	<i>25</i>
<i>Figura 19 – Ecrã de pesquisa de utilizadores</i>	<i>26</i>
<i>Figura 20 – Ecrã de configurações</i>	<i>27</i>
<i>Figura 21 – Ecrã de correspondências.....</i>	<i>28</i>
<i>Figura 22 – Ecrã de pesquisa de utilizador, com um resultado</i>	<i>29</i>
<i>Figura 23 – Ecrã de configurações, com configurações de pesquisa alteradas</i>	<i>30</i>
<i>Figura 24 – Ecrã de pesquisa de utilizadores, ilustrando a ação de aprovação de um utilizador</i>	<i>31</i>
<i>Figura 25 – Ecrã de login, ilustrando um erro de autenticação</i>	<i>32</i>
<i>Figura 26 – Ecrã de configurações, ilustrando a gravação de um novo áudio pessoal.....</i>	<i>33</i>
<i>Figura 27 – Ecrã das pesquisas, mostrando uma mensagem de correspondência</i>	<i>34</i>
<i>Figura 28 – Ecrã de correspondências.....</i>	<i>34</i>
<i>Figura 29 – Ecrã de chat, com uma mensagem de áudio</i>	<i>35</i>
<i>Figura 30 – Ecrã de chat, com interação inicial áudio entre dois utilizadores</i>	<i>36</i>
<i>Figura 31 – Ecrã de chat, com interação de áudio e de texto</i>	<i>36</i>

<i>Figura 32 – Ecrã de chat, ilustrando a opção de apagar a correspondência.....</i>	<i>37</i>
<i>Figura 33 – Ecrã de correspondências, sem nenhuma correspondência</i>	<i>38</i>

1. Introdução

1.1 Enquadramento

Desde o seu surgimento que as redes sociais se têm tornado cada vez mais populares e indispensáveis no quotidiano de qualquer um. São um mundo aberto de partilha de experiências e opiniões que alcançam tanto os amigos mais chegados como completos desconhecidos que, ao terem algo em comum, se poderão conectar e criar ligações pessoais.

Contudo, toda esta abertura e expansão no mundo virtual acabou por ter algum impacto negativo no mundo real. Existe uma tendência para que as pessoas se isolem e reservem mais do mundo exterior, passando de extremamente sociais virtualmente a antissociais ao vivo. A adaptação às redes sociais fez com que todos se preocupassem cada vez mais com o que partilham, filtrando, editando e preparando previamente o conteúdo antes de o publicar tornando as suas vidas irreais e enfatizadas, no entanto, muito desejadas. Todos estes fatores tornaram os utilizadores dependentes da aprovação generalizada dos outros nessas mesmas redes, fechando-se mais atrás de um ecrã e isolando-se mais do mundo, tendo assim uma menor capacidade de estabelecer relações não virtuais com os outros.

Além disso, mesmo que tentemos abrir os horizontes e conhecer pessoas novas de forma convencional, torna-se uma tarefa complicada sendo que, com o *stress* e falta de tempo livre em vida adulta, não nos sobra grande margem de manobra para além de conviver com o círculo de amigos de sempre, família e colegas de trabalho.

Apesar de com o avanço tecnológico e da internet dispormos de ferramentas que nos permitem falar com quem quisermos e de que, atualmente, alguns preconceitos terem sido transpostos, existe ainda um certo receio de nos tentarmos conectar com novas pessoas por esses meios e pormos a nossa segurança em risco.

Tendo em conta o acima referido começaram a surgir aplicações que permitem facilitar a interação entre pessoas desconhecidas através, por exemplo, de um simples toque no telefone, criando uma conexão sem receios nem compromissos.

1.2 Situação Atual

Como já referido, as redes sociais tornaram-se extremamente populares e procuradas, o que provocou a constante evolução e inovação das mesmas.

Posteriormente apareceram aplicações mais direcionadas para encontros e relações amorosas que permitem conhecer novas pessoas, tais como o *Tinder* [1], *Happn* [2] ou *Badoo* [3].

Contudo, todas elas assentam na mesma forma de iniciar uma conexão, permitem os utilizadores criarem perfis com fotos e descrições que facilitem a decisão de eventuais interessados em iniciar uma conversa. Apesar desse método ter ganho grande popularidade e adesão, a decisão de poder haver compatibilidade ou não com alguém depende unicamente da apreciação visual e física, tornando assim a experiência mais superficial, pouco realista e perdendo a essência e espontaneidade de conhecer alguém ao acaso.

Outra tendência mais recente no mundo das redes sociais é o envio de mensagens de áudio em vez de texto, visto ser uma forma mais prática e rápida de partilhar uma ideia ou informação com alguém. Além disso, torna-se numa interação mais pessoal pois, ao contrário das mensagens escritas, não são colocados filtros e a mensagem é recebida sem barreiras, com a emoção do momento como se estivesse a haver um diálogo pessoalmente.

Tendo em conta estas conclusões, o projeto tem como objetivo o desenvolvimento de uma aplicação que permita conhecer novas pessoas através do mundo virtual, mas tendo uma experiência mais pessoal, sem julgamento visual.

1.3 Objetivos

O projeto *Fowyv, Find Others With Your Voice*, tem como objetivo a disponibilização de uma aplicação móvel que permita aos seus utilizadores conhecer novas pessoas, mas desta vez através de uma interação inicial de áudio.

Cada perfil possui um ficheiro de áudio pessoal visível a todos e será a partir desse recurso que os restantes utilizadores escolhem se querem conhecer melhor essa pessoa ou se, pelo contrário, não lhes desperta curiosidade para mais. Se ambos os interlocutores declaram interesse em se conhecerem podem desenvolver uma conversa numa caixa de diálogo, mas inicialmente só por mensagens de áudio de forma a criar uma experiência mais pessoal, mas, após um determinado número de interações, a aplicação permitirá o envio de mensagens de texto.

1.4 Organização do documento

O restante relatório encontra-se organizado da seguinte forma:

- Capítulo 2 – São apresentadas as funcionalidades a implementar no projeto e temas a ter em consideração na elaboração do mesmo;
- Capítulo 3 – É descrita a solução adotada, a sua arquitetura, a sua descrição detalhada como as tecnologias usadas e a disponibilização da mesma;
- Capítulo 4 – É apresentada a avaliação experimental da solução proposta.
- Capítulo 5 – São apresentadas as conclusões.

2. Formulação do Problema

Neste capítulo são abordados os principais temas a ser tidos em conta na estruturação do projeto de forma a dar resposta aos objetivos propostos anteriormente. A secção 2.1 descreve a terminologia utilizada. Na secção 2.2 são descritos os requisitos a ter em consideração no planeamento do projeto. É na 2.3 são apresentados os casos de uso da aplicação na ótica do utilizador. Os requisitos técnicos são retratados na secção 2.4.

2.1 Terminologia

Renderizar - Ato de compilar e obter o produto final de um processamento digital.

2.2 Requisitos

Tendo em conta o objetivo enunciado no capítulo anterior, surgiram os requisitos abaixo enumerados que foram tidos em conta para a elaboração deste projeto.

2.2.1 Comunicação tempo real

Com o propósito de criar uma aplicação que permita interação entre utilizadores, a implementação da mesma deve ter em vista permitir uma comunicação constante, que notifique no mínimo tempo possível os utilizadores de eventos despoletados por outros.

2.2.2 Simplicidade e Intuitiva

A aplicação de ser simplista e intuitiva de forma a que o utilizador seja levado a interagir constantemente e não perca rapidamente o foco ou interesse.

2.2.3 Armazenamento de dados

No armazenamento dos dados é necessário ter em conta que este tipo de projeto gera uma grande quantidade de informação que não requer a execução de grandes transações nem de lidar com *queries* complexas para obtenção da informação, necessitando de se focar mais na rapidez e escalabilidade do armazenamento.

É necessário refletir na geração/utilização de ficheiros de dimensões consideráveis, como áudios, que necessitam de ser armazenados em locais com grande capacidade de alojamento e que permitam um rápido acesso aos mesmos.

2.2.4 Versatilidade

Tendo este projeto como objetivo a disponibilização de uma aplicação móvel, deve ser tido em conta a versatilidade da mesma de forma a que as especificações dos dispostos não sejam limitadoras para a utilização da mesma.

2.2.5 Escalabilidade

É também importante ter em mente o desenvolvimento de um sistema que permita uma rápida escalabilidade do mesmo se assim for necessário.

2.3 Descrição da aplicação

No âmbito deste projeto foram definidas as quatro ações principais na experiência do utilizador (ilustrados na fig.1): permitir ao utilizador fazer *login*, ouvir os áudios de outros utilizadores, interagir com correspondências e permitir configurar a sessão.

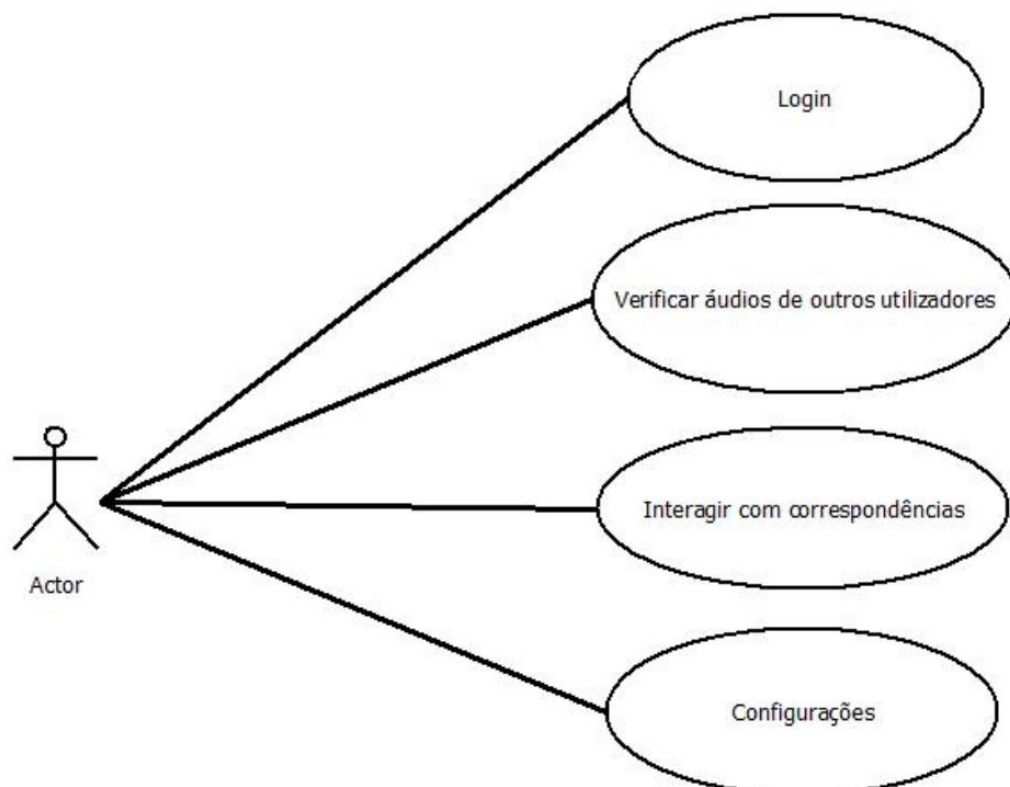


Figura 1 - Casos de Uso considerados para a experiência dos utilizadores

2.3.1 Autenticação do utilizador

O *login* representa o processo de utilização inicial de acesso à aplicação, apresentado na figura 2.

O usuário terá sempre de se autenticar para poder utilizar a aplicação, criando uma conta de utilizador que lhe permita posteriormente se autenticar utilizando as credenciais que forneceu.

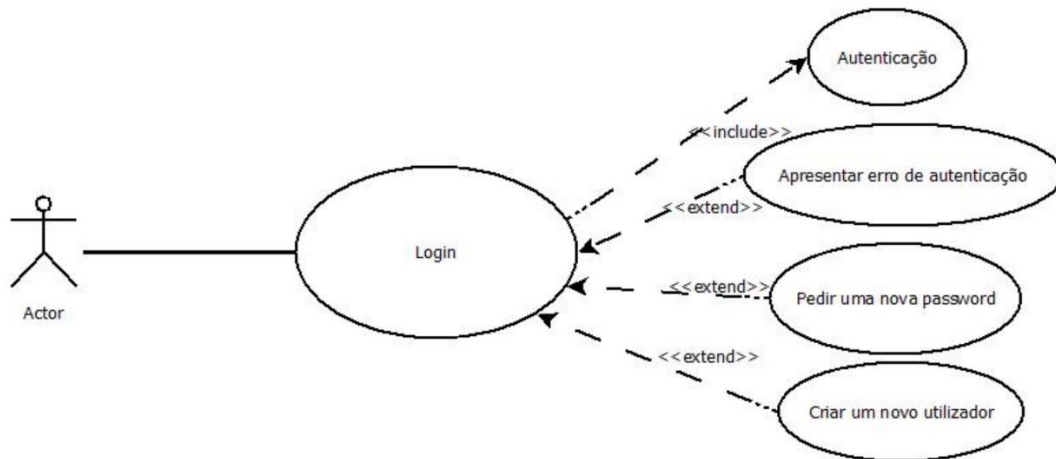


Figura 2 – Use case Login

- Autenticação:
 - O usuário terá de fornecer as credenciais que forneceu na criação da sua conta de utilizador ou poderá não necessitar de as introduzir se já as forneceu previamente.
- Apresentar erro de autenticação:
 - Nesta eventualidade, será apresentado ao utilizador uma mensagem de erro referenciando qual o erro que terá ocorrido.
- Pedir uma nova password:
 - Será dada a possibilidade ao utilizador de poder pedir uma nova palavra-passe.
- Criar um utilizador:
 - Neste ponto, o utilizador poderá introduzir as características que deseja que fiquem associadas ao seu perfil de utilizador, assim como gravar o áudio que ficará a si associado.

2.3.2 Verificar áudios de outros utilizadores

Nas configurações existe a possibilidade de editar preferências como a margem de idades e o género que se procura, ou usar as preferências *default*. Tendo essa informação em conta, serão apresentados áudios de diferentes perfis previamente filtrados e após o utilizador ouvir o conteúdo, poderá decidir se aprova e pretende conhecer mais dessa pessoa (movendo o ícone do áudio para direita no ecrã) ou se a descarta (movendo o ícone do áudio para a esquerda), como esquematizado na figura 3.

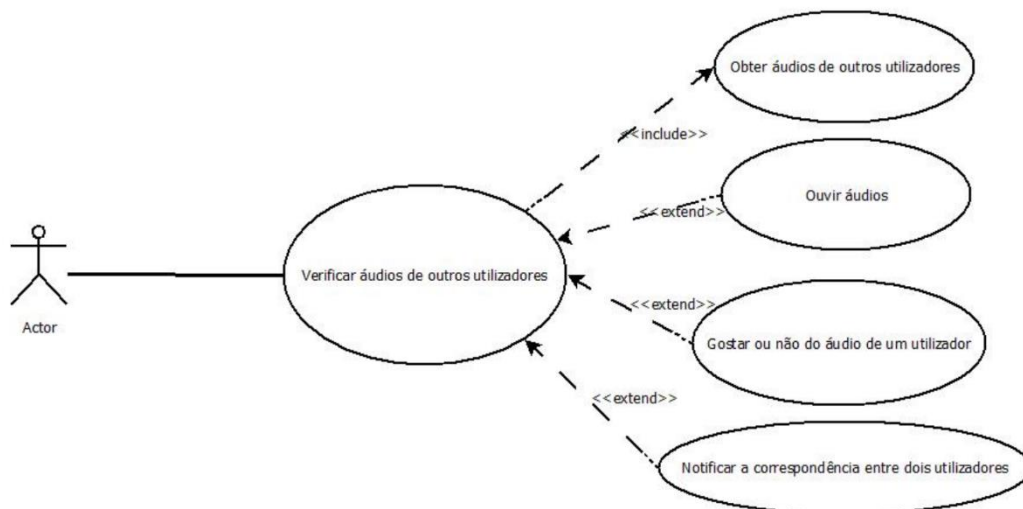


Figura 3 – Use case Verificar áudios de outros utilizadores

- Obter áudios de outros utilizadores:
 - Serão tidas em conta as características que o utilizador definiu para a procura de áudios a apresentar.
- Ouvir áudios:
 - Neste ponto, o utilizador poderá escutar os áudios gravados pelos outros utilizadores quando desejar.
- Gostar ou não do áudio de um utilizador:
 - Será dada a escolha ao utilizador de sinalizar se gostou ou não do áudio que escutou.
- Notificar a correspondência entre dois utilizadores:
 - Na eventualidade de dois utilizadores sinalizarem que gostaram do áudio um do outro, será apresentado ao último utilizador a interagir uma mensagem da correspondência efetuada e a opção de começar a interagir logo com o outro utilizador.

2.3.3 Interagir com correspondências

Ao interagir para procurar correspondências, o utilizador poderá verificar todos os perfis com quem partilhou interesse mútuo, tanto os perfis com quem interagiu anteriormente como também os novos perfis com quem fizeram combinação de aprovações, representado na figura 4. Pode interagir através de um *chat* onde, dependendo do nível da conversa, poderá usufruir de diferentes meios de comunicação como texto, áudio, imagem e vídeo.

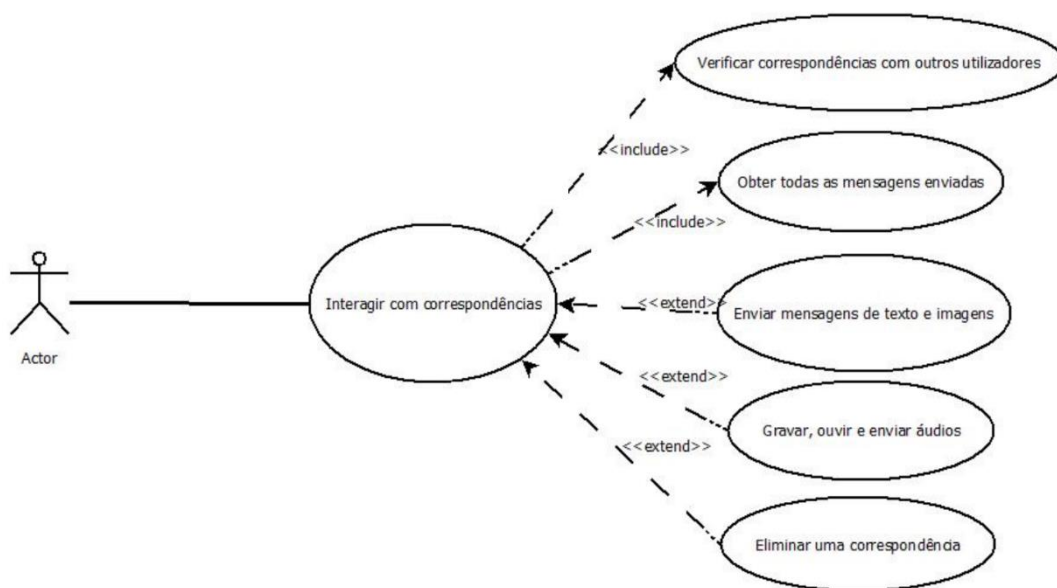


Figura 4 – Use case Interagir com correspondências

- Verificar correspondência com outros utilizadores:
 - Irá sempre ser efetuada uma pesquisa pelas correspondências do utilizador, verificando se existe novos perfis ou se algum foi removido.
- Obter todas as mensagens enviadas:
 - É conveniente verificar sempre a caixa de entrada do chat para atualizar as conversas dos utilizadores e confirmar se tem novas mensagens por abrir.
- Enviar mensagens de texto e imagens:
 - Será dada a possibilidade ao utilizador de poder enviar mensagens de texto, assim como imagens.
- Gravar, ouvir e enviar áudios:
 - Neste ponto, um utilizador poderá gravar, ouvir e enviar mensagens de áudio, assim como ouvir as mensagens de áudio do outro utilizador.
- Eliminar uma correspondência:
 - Na eventualidade de um utilizador não pretender continuar a interação com uma correspondência poderá eliminá-la, permitindo assim que termine a interação entre ambos.

2.3.4 Configurações

Neste ponto de utilização, exposto na figura 5, o utilizador poderá configurar determinados aspetos do funcionamento da aplicação como os parâmetros de pesquisa de áudios de outros utilizadores, assim como poderá alterar características relativas ao seu perfil.

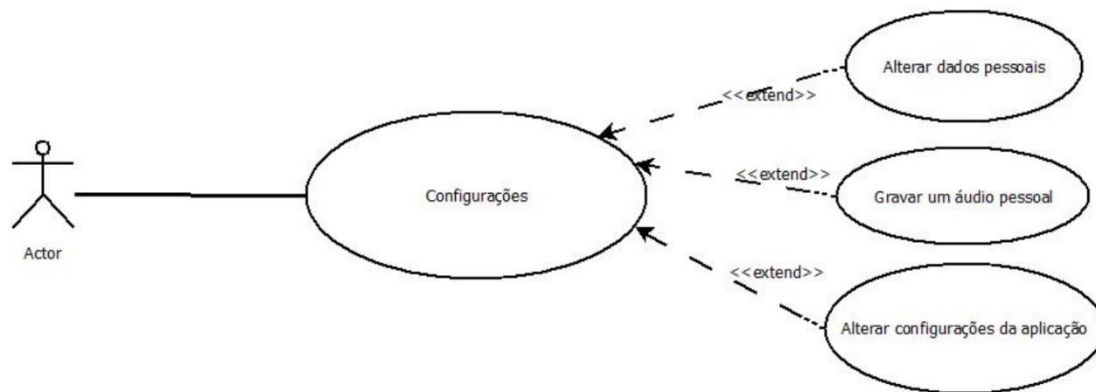


Figura 5 – Use case Configurações

- Alterar dados pessoais:
 - O utilizador poderá alterar dados relativos à sua conta como, por exemplo, o nome que ficará visível para os outros utilizadores.
- Gravar um áudio pessoal:
 - Será neste ponto que o utilizador poderá alterar o áudio associado à sua conta, gravando uma nova versão.
- Alterar configurações da aplicação:
 - Poderão ser alteradas determinadas configurações da aplicação que estejam associadas ao utilizador como, por exemplo, a margem de idade dos utilizadores a procurar.

2.4 Tecnologias utilizadas

2.4.1 Aplicação móvel

Para o desenvolvimento da aplicação móvel, foi decidido usar a *framework* de *Javascript React Native* [4], sobretudo pela sua versatilidade que permite desenvolver aplicações móveis tanto para *Android* [5] como para *iOS* [6]. Apesar que o foco deste projeto assenta no desenvolvimento da aplicação móvel para *Android*, será importante deixar abertura para futuramente ser possível implementá-lo em *iOS*.

A existência da abstração da parte nativa dos dispositivos móveis facilita a adaptação e o desenvolvimento da aplicação, para além de que a existência de uma vasta comunidade que desenvolve sobre esta tecnologia facilita o encontro de diversas bibliotecas que satisfazem as necessidades para implementar as funcionalidades desejadas.

2.4.2 Aplicação servidora

Neste caso foi escolhido que a implementação da mesma deve assentar sobre uma aplicação servidora *Javascript* que se executada sobre *Node.js* [7], necessariamente por ter uma grande versatilidade devido à comunidade que desenvolve sobre a mesma permitir uma rápida escalabilidade e permitir expor as funcionalidades necessárias à aplicação móvel verificando exemplos de aplicações do mesmo tipo.

2.4.3 Base de dados

Como referido na secção anterior, os fatores a ter mais em conta na escolha da tecnologia da base de dados deveriam ser a rapidez e escalabilidade da mesma. Portanto, foi escolhido usar *MongoDB* [8] como base de dados, pois é uma base de dados não relacional com estabilidade comprovada e com grande utilização com aplicações *Node.js* na atualidade, facilitando a sua integração neste projeto.

2.4.4 Serviço de armazenamento

Para alojar os ficheiros de multimédia da aplicação foi decidido usar o serviço *Amazon Simple Storage Service (Amazon S3)* [9], pois é um serviço bastante utilizado atualmente para o armazenamento de diverso tipo de ficheiros através de aplicações web, permitindo alojar ficheiros até 5 Terabytes e que inclusivo disponibiliza uma biblioteca de fácil de integração e utilização para projetos Node.

3. Solução Proposta - Grandes Ideias

3.1 Arquitetura do Sistema

Na arquitetura do sistema proposto existem quatro componentes principais, como mostra a figura 6. A aplicação móvel será o ponto de interação único do utilizador com toda a aplicação, expondo todas as funcionalidades definidas na secção anterior.

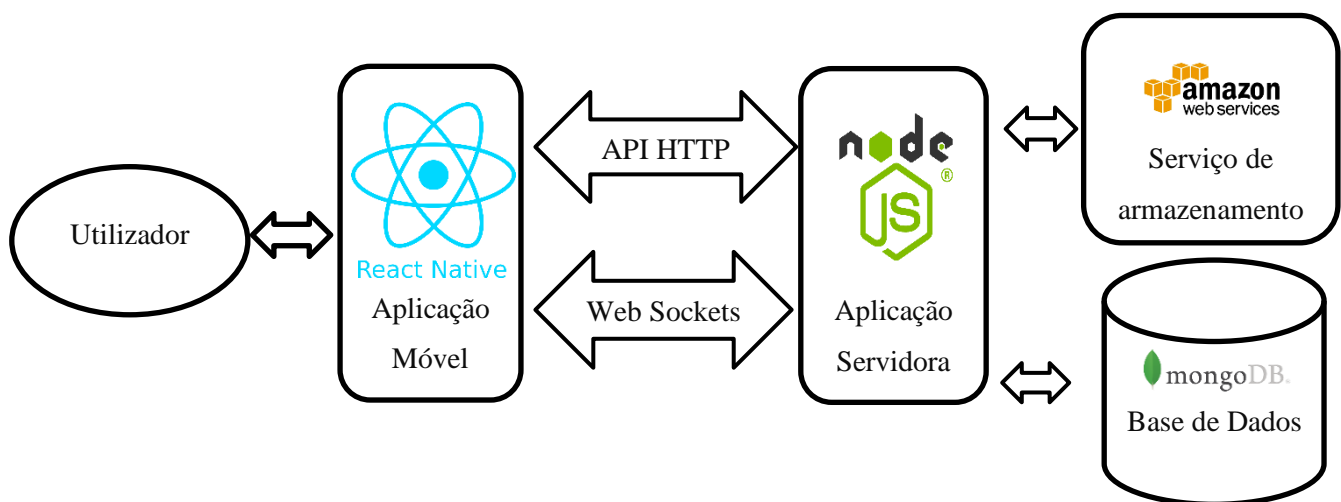


Figura 6 - Arquitetura do sistema proposto

A aplicação móvel tem como objetivo permitir ao utilizador criar uma conta, gravar áudios e comunicar com os outros utilizadores. Esta irá comunicar com a aplicação servidora para guardar/obter dados e estabelecer conversas através de uma *API HTTP* e *WebSockets*.

A função da aplicação servidora será expor a *API HTTP* a ser consumida pela aplicação móvel, nomeadamente para autenticação, guardar e expor dados. Também permite a criação de *WebSockets* que são uteis para estabelecer uma conexão constante viável para a troca constante de informação resultante das interações entre utilizadores.

Para o armazenamento de informação, ficou decidido separar os ficheiros multimédia gerados/utilizados na aplicação num serviço de alojamento de ficheiros do resto da informação gerada no decorrer da execução da aplicação que ficaria numa base de dados separada.

3.2 Descrição detalhada da solução

Como se pode analisar na secção da arquitetura da solução, a parte funcional do projeto assenta na aplicação móvel e na aplicação servidora, que comunicam entre si através da *API HTTP* exposta, assim como através de *WebSockets*. Na figura 7 é demonstrado o esquema sequencial de um exemplo de comunicação *HTTP* entre ambas as aplicações.

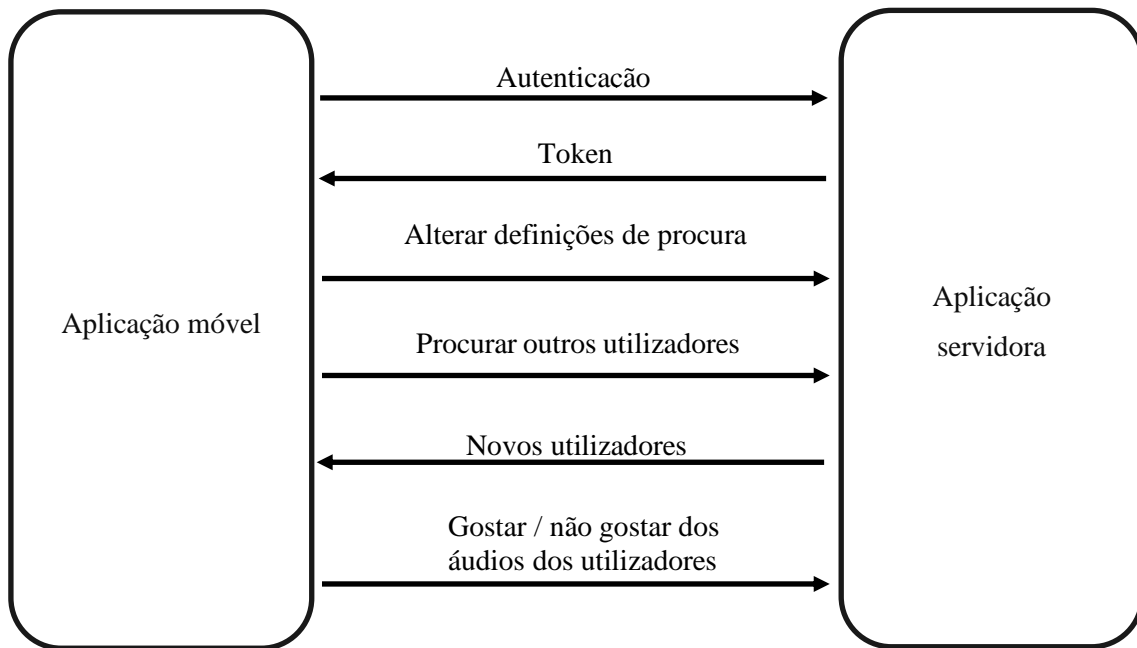


Figura 7– Esquema sequencial da comunicação entre a aplicação móvel e a aplicação servidora

A *API* exposta pela aplicação servidora permite à aplicação móvel guardar e obter os dados originados pela sua utilização, assim como disponibiliza as funcionalidades de criação e autenticação dos utilizadores tanto necessárias para poder utilizar a aplicação, como para consumir os outros diversos *endpoints* da *API*. É através dela que é possível procurar novos utilizadores ou alterar as configurações de pesquisa dos mesmos.

Na figura 8 é demonstrado um fluxo de comunicação usando os *websockets* entre as duas aplicações.

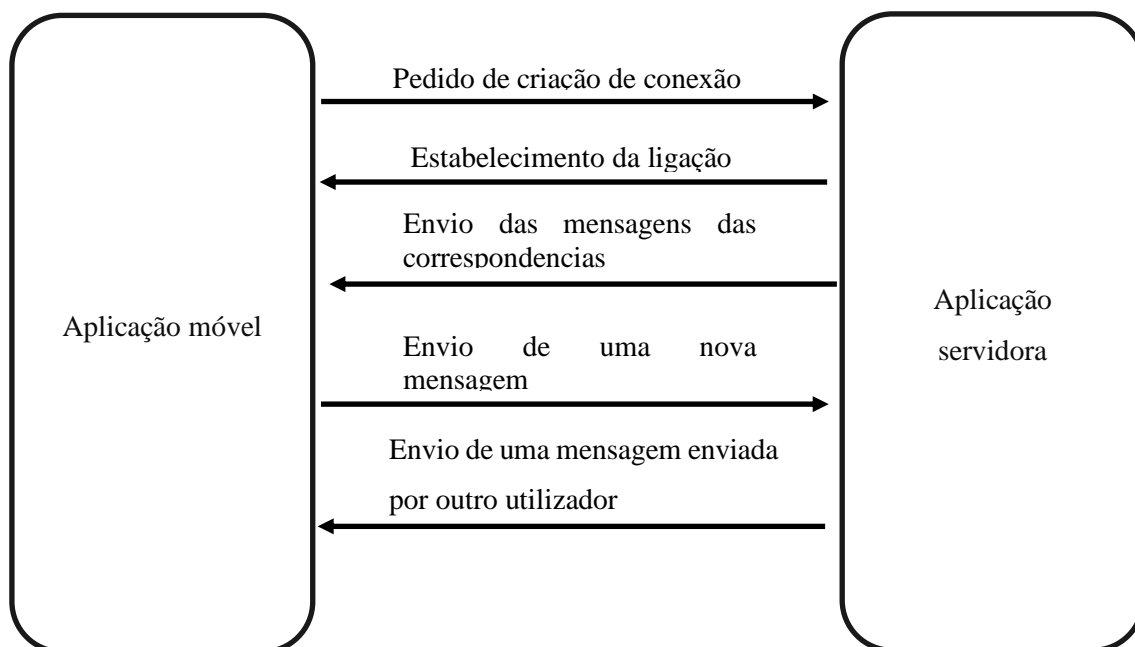


Figura 8 – Esquema sequencial de comunicação através de websockets entre a aplicação móvel e a aplicação servidora

A comunicação através de *websockets* tem com propósito permitir uma conexão *full-duplex* entre ambas as aplicações, suportando a troca de comunicação constante resultante das interações entre os utilizadores e tentando evitar a necessidade de *pooling* para manter a aplicação atualizada.

Quando um utilizador deseja enviar uma mensagem para outro utilizador, é emitido através da aplicação móvel um dos diversos eventos que a aplicação servidora terá subscrito para ser notificada no início da criação da conexão para escutar. Após esse evento ser recebido, o mesmo é processado e a informação obtida da mensagem é armazenada na base de dados, caso a mensagem possua ficheiros áudio serão também armazenados no serviço de alojamento.

Posteriormente ao processamento da mensagem, se o outro utilizador também tiver a aplicação móvel aberta e com uma conexão *websocket* estabelecida à aplicação servidora, esta notificará o utilizador através de um dos eventos que a aplicação subscreeveu no início da criação da conexão para que a aplicação móvel seja notificada.

Após o processamento da informação recebida através dos pedidos *HTTP* ou dos eventos *websockets*, a mesma é armazenada na base de dados e, em certas ocasiões, no serviço de

alojamento de ficheiros. Na figura 9 é demonstrado um exemplo do armazenamento de uma mensagem de áudio.

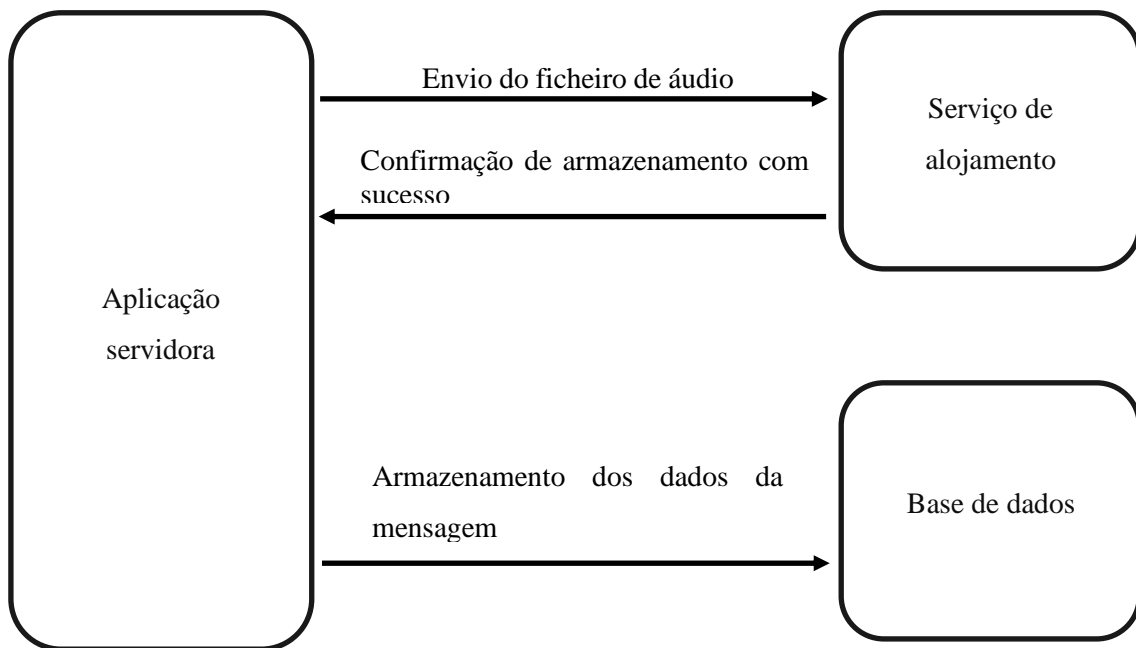


Figura 9 – Esquema sequencial do armazenamento de uma mensagem de áudio

O armazenamento da informação resultante da execução da aplicação móvel é feito na base de dados, desde informação relativa à conta do utilizador, até ao histórico de interações entre dois. O serviço de alojamento tem como propósito permitir o armazenamento dos ficheiros áudio que são enviados nas mensagens.

3.2.1 Aplicação móvel

A aplicação cliente foi desenvolvida utilizando a *framework React Native* que assenta no desenvolvimento de componentes *UI* reutilizáveis que possam ser compilados para componentes *UI* nativos.

Os componentes em *React Native* podem ser classificados como funções que recebem um *input* que afeta um estado guardado dentro da função. Esse mesmo estado é tido em conta para *renderizar* algo.

A figura 10 que se segue demonstra como se encontra estruturada a aplicação móvel.

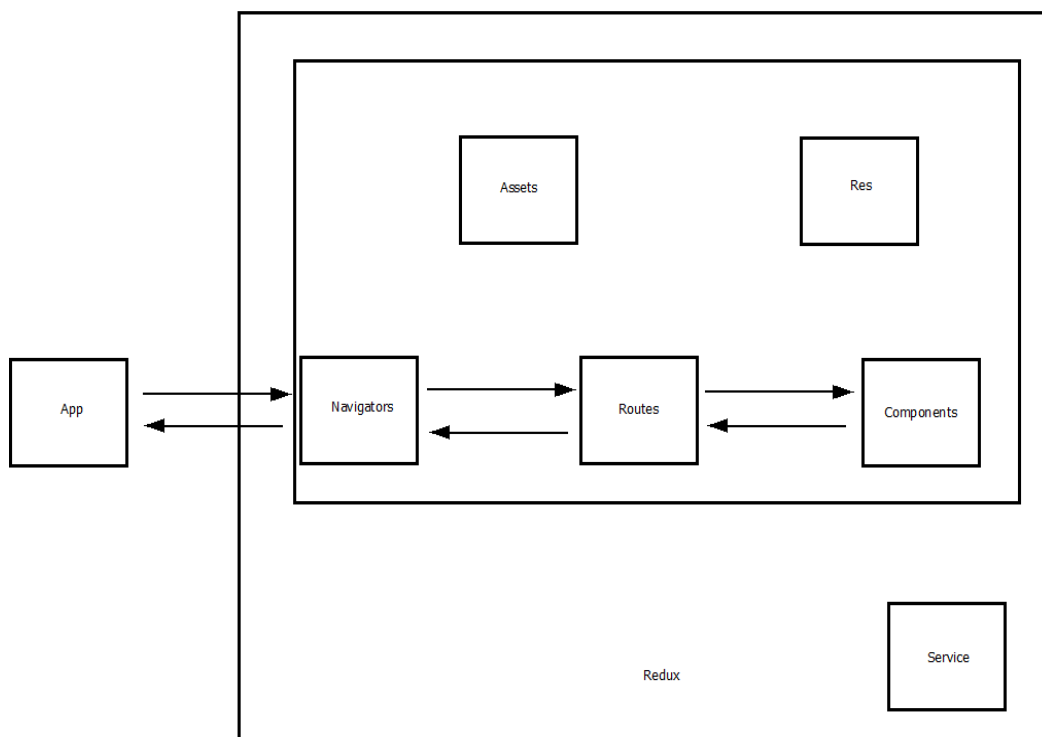


Figura 10 -Estrutura da aplicação móvel

A aplicação encontra-se assente na ideia de um *stack* global de navegação, exposto no componente *RootNavigatorContainer* utilizando a biblioteca *react-navigation*, que facilita a gestão dos ecrãs. Neste componente, estão definidos todos os *stacks* que são possíveis de navegar na utilização da aplicação: o *AuthenticationStackNavigator* responsável por lidar com os ecrãs de autenticação, o *MainStackNavigator* onde se encontram definidos todos os ecrãs possíveis de navegar por um utilizador autenticado, com exceção do ecrã de chat, que foi decidido colocar no *ChatStackNavigator* pois todos os ecrãs definidos no *MainStackNavigator* têm em comum diferentes aspetos de transição de ecrã, assim como a barra de navegação no topo dos seus ecrãs, aspetos que o ecrã de chat não partilha.

Um utilizador, ao entrar na aplicação será redirecionado para o *AuthenticationStackNavigator*, primeiramente para o ecrã *Home* durante um curto espaço de tempo, visualizando um ecrã de apresentação da aplicação, sendo posteriormente redirecionado para o ecrã *Login* onde poderá introduzir as suas credenciais para se autenticar, ou navegar para o ecrã *NewAccount* para iniciar a criação de uma nova conta.

Posteriormente, o utilizador define o seu perfil no ecrã *SetProfile*, grava um ficheiro de áudio pessoal através da interação com o componente *PersonalAudioContainer* usado neste ecrã, este permite renderizar o componente *PersonalAudioRecorder*, que é um componente modal que se sobrepõem sobre o ecrã atual. Por fim, será possível gravar um áudio e escutá-lo, utilizando os componentes *AudioRecorder* e *AudioPlayer*, e as bibliotecas *react-native-audio* e *react-native-sound*.

Estando um utilizador autenticado, será reencaminhado para o *stack MainStackNavigator*, onde visualizará o ecrã *ListenLobby* e poderá verificar o resultado da pesquisa de novos utilizadores com os quais ainda não interagiu, podendo escutar os áudios de cada perfil, deslizar para a esquerda se não quiser obter uma correspondência com o utilizador ou para a direita em caso contrário. Neste ecrã existe a possibilidade de navegar para os ecrãs *Settings* e *MatchLobby* através da barra de navegação definida no componente *NavBar*, comum aos três ecrãs que constituem o *MainStackNavigator*.

Ao visualizarmos o ecrã *MatchLobby*, podemos verificar todas as correspondências de um utilizador, sendo cada uma delas definida através do componente *Match*, permitindo saber o nome e o ícone associado ao utilizador correspondente. Ao clicar sob qualquer uma das correspondências, a aplicação irá *renderizar* o ecrã *Chat*, permitindo o utilizador poder interagir com a correspondência através do envio de mensagens de áudio e de texto, assim como visualizar o histórico de interação. As mensagens de áudio gravadas podem ser vistas e escutadas no histórico de interação através do componente *AudioMessage*. É dada a possibilidade ao utilizador de voltar ao ecrã dos *MatchLobby*.

O utilizador também pode navegar para o ecrã de *Settings*, onde poderá alterar configurações da pesquisa de novos utilizadores, assim como visualizar os seus dados pessoais ou gravar um novo ficheiro de áudio pessoal.

Para facilitar a gestão e partilha de estado entre componentes, é utilizada uma *store* global que é gerida utilizando a biblioteca *javascript Redux*, permitindo que qualquer componente da aplicação possa alterar o estado global através das *actions*, permitindo posteriormente especificar as alterações a fazer ao estado da aplicação através dos *reducers*. É também através das *actions* que é feita a comunicação com a *API* exposta pela aplicação servidora através do componente *API*, definido na diretoria *service*, para guardar e obter dos dados da aplicação.

Também na aplicação móvel foi utilizada a biblioteca *socket.io-client* para permitir comunicar com a aplicação servidora através de *websockets* para garantir uma ligação constante que permita à aplicação cliente subscrever eventos emitidos pela aplicação servidora, assim como emitir eventos que serão subscritos pela mesma.

Na criação das conexões *websocket*, é passado o *token* gerado na autenticação do utilizador, de forma a garantir que a utilização das mesmas é feita apenas por utilizadores que tenham *tokens* e que sejam válidos.

As conexões são utilizadas relativamente ao envio de mensagens de texto ou áudio, a alterações na base de dados como o envio de novas mensagens de outros utilizadores ou o envio de ficheiros de áudio, não tendo de estar constantemente a fazer *polling* para tentar obter novos dados ou a fazer pedidos constantes para o envio de mensagens do utilizador.

3.2.2 Aplicação Servidora

API HTTP

A *API HTTP* é exposta através de uma aplicação *web* que assenta sobre a *framework Express* para ser executada no ambiente de execução multiplataforma *NodeJS*, expondo a *API* a ser consumida pela aplicação cliente.

A figura 11 que se segue demonstra como se encontra estruturada a aplicação, em relação à *API HTTP*.

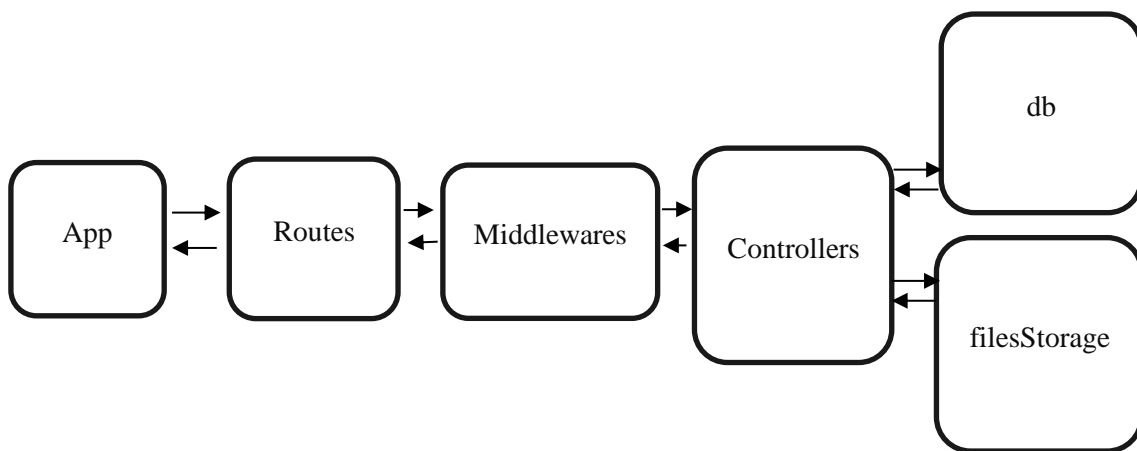


Figura 11 - Estrutura da aplicação servidora para a exposição da *API HTTP*

A aplicação servidora define uma *API HTTP* através de uma estrutura de *middlewares* que são declarados no *entry point app*. As *routes* são os *middlewares* declarados na *app* para definir os *endpoints* que se encontram expostos, onde cada *route* por sua vez define quais os *Middlewares* que o pedido deve passar para ser considerado válido. Após ser validado o pedido, este é redirecionado para os *controllers* que estão encarregues de comunicar com a camada de dados.

A aplicação servidora permite a autenticação do utilizador através do envio das credenciais do mesmo para um *endpoint* que posteriormente as valida com a informação que contem na base de dados. Posteriormente à validação, é gerado um *JSON Web Token*, facultando ao utilizador um *token* com um tempo de expiração, de forma a permitir que este possa poder consumir serviços expostos pela aplicação servidora, tanto a *API HTTP*, como os *WebSockets*.

Websockets

A aplicação permite a criação de conexões utilizando o protocolo *websocket* para estabelecer uma ligação *full-duplex* através da utilização da biblioteca *socket-io* entre a aplicação cliente e a aplicação servidora.

Na figura 12 é demonstrada a estrutura assente na aplicação servidora para permitir a ligação de *websockets*.

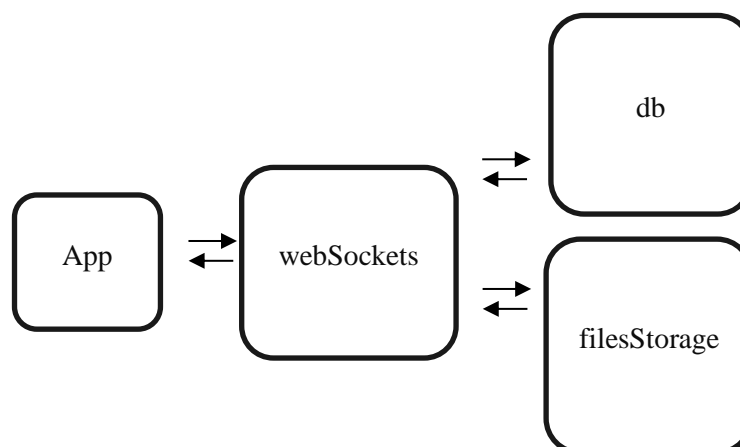


Figura 12 - Estrutura da aplicação servidora para a exposição dos websockets

As conexões *websocket* tem como propósito viabilizar a comunicação em tempo real entre os clientes, permitindo que o cliente envie notificações para a aplicação servidora e a mesma o notifique, sem que haja a necessidade de estar constantemente a fazer *polling* para a troca de novos dados sobre a interação entre utilizadores.

A utilização dos *websockets* tem como forma de autenticação a utilização do *JWT* passado na criação da conexão, que foi fornecido pela *API HTTP*, posteriormente validando o mesmo e autenticando os eventos recebidos.

A aplicação servidora, ao estabelecer uma nova conexão *websocket* define os eventos que são responsáveis pela obtenção e envio das mensagens da interação dos utilizadores, assim como é responsável pela atualização das mensagens nos clientes, enviando notificações sempre que existem alterações relativas à interação entre os utilizadores.

Após ser estabelecida uma ligação *websocket*, a referência para a mesma é guardada do lado da aplicação servidora, com o próximo de saber quantas conexões existem e quem é que se encontra conectado.

Quando chega um pedido de criação de conexão, são subscritos os eventos que a aplicação servidora irá ficar à escuta que sejam emitidos pela aplicação cliente, como o envio de *receiveMessage* ou *uploadMessage*. Por sua vez a aplicação servidora emite eventos para a aplicação cliente quando tem uma mensagem a enviar ao utilizador.

O envio e receção de mensagens através de *websockets* é definido através do envio de objetos *JSON* que contêm as informações relativas ao tipo de mensagem que é, podendo ser de texto ou áudio, diferenciando que nas mensagens de áudio as mesmas contêm o conteúdo dos ficheiros áudio que são recebidas e enviadas como *strings* codificadas BASE64, para além do tipo de ficheiro, de forma a poder alojar no serviço de alojamento de ficheiros.

Posteriormente à receção das mensagens, as mesmas são guardadas na base de dados, sendo o tipo de mensagem tido em conta para verificar se é uma mensagem de áudio, fazendo upload ou download do conteúdo dos ficheiros de áudio no serviço de alojamento de ficheiros, dependendo da ação.

Comunicação com a base de dados

A comunicação com a base de dados é garantida pela utilização da biblioteca *mongodb* que permite estabelecer ligações com base de dados *MongoDB*.

A utilização da base de dados assenta na implementação exposta no ficheiro *dbCliente*, expondo funções que permitem criar ligações à base de dados e executar desde *queries* unitárias, até transações para executar *queries* extensas, que são passadas pelos outros módulos da aplicação aquando a utilização destas funções.

Também é disponibilizada nesta implementação a criação de *change streams*, que é abordado no capítulo abaixo referente à implementação da base de dados.

Esta funcionalidade é utilizada de forma a perceber quando tenha havido alterações do histórico de interação entre dois utilizadores. Através da inserção de uma nova mensagem na base de dados, é possível verificar o último utilizador a quem se destina a mensagem e verificando as conexões *websocket* abertas, é possível verificar se o utilizador a receber a mensagem encontra ativo e se deve receber uma notificação com a nova mensagem.

Comunicação com o serviço de alojamento

A comunicação com o serviço de alojamento é garantida pela biblioteca *aws-sdk* que permite fazer *upload* e *download* de conteúdo para o *bucket* explícito, abstraindo da implementação necessária para a utilização do serviço Amazon S3 e facilitando o uso da mesma.

3.2.3 Base de dados

A base de dados utilizada é *MongoDB*, trata-se duma base de dados não relacional onde o armazenamento de dados é feito em documentos com uma estrutura interna semelhante a *JSON*, sendo possível guardar esses mesmo em coleções.

A base de dados encontra-se definida através de 5 coleções, podendo se verificar na figura 13 a estrutura dos documentos que se encontram em cada uma.

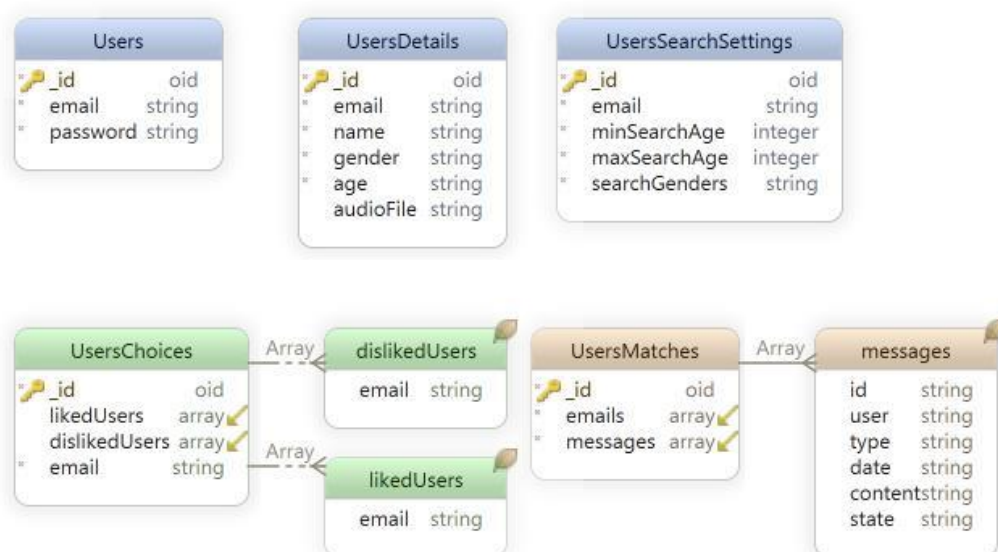


Figura 13 - Estrutura dos documentos das coleções

Na coleção *Users* é guardada informação dos utilizadores como os seus emails e suas passwords. Quanto à coleção *UsersDetails*, são guardadas aqui as informações relativas aos detalhes de cada utilizador. Quanto à *UsersSearchSettings*, contém informação relativa às preferências de pesquisa de novos utilizadores. A *UsersChoices* é onde são armazenadas as informações sobre os perfis aprovados e não aprovados pelo utilizador. Por fim, a coleção *UsersMatches* contém informação relativa às correspondências ocorridas, incluindo também o histórico de interação dos mesmos.

Uma das vantagens da utilização de *MongoDB* nesta aplicação, é a utilização de *change streams*, referidas na secção acima, que consistem em conexões que permitem à aplicação servidora ser notificada aquando uma coleção foi alvo de uma operação, quer seja de inserção, alteração ou até remoção. Tendo em conta a informação e sabendo as conexões que se encontram abertas, é fácil notificar os utilizadores de alterações que possam ter surgido.

Com isso, é fácil verificar aquando a adição de uma nova mensagem ao histórico de interação entre dois utilizadores, podendo então notificar o outro utilizador da nova mensagem enviada.

3.2.4 Serviço de alojamento de ficheiros

O serviço de alojamento *AWS3* foi utilizado neste projeto com o propósito de assegurar o armazenamento de ficheiros gerados durante a execução da aplicação móvel, sendo neste caso ficheiros áudio, em um local seguro e de fácil acesso à aplicação servidora.

A utilização do serviço de alojamento é possível através da utilização dos módulos fornecidos pela própria *Amazon* para comunicação entre o serviço de alojamento e aplicações servidora, como neste caso *NodeJS*, que foi referido no capítulo acima.

O serviço de alojamento consiste no armazenamento de ficheiros em *Buckets* que são contentores que permitem alojar objetos, sendo esses objetos definidos por conteúdo e por *metadados*, onde é possível através destes últimos saber os detalhes dos ficheiros como o nome ou última data em que foi alterado.

O nome dos ficheiros guardados é um *guid* gerado pela aplicação servidora aquando o *upload* do conteúdo dos ficheiros para o serviço de alojamento. Este mesmo nome é guardado nas mensagens de áudio guardadas no histórico de interação entre correspondências para que posteriormente a aplicação móvel possa enviar um pedido de *download* para a aplicação servidora dos ficheiros.

Ao fazer *upload* dos ficheiros, os mesmos são guardados e encriptados do lado do serviço utilizando o algoritmo *AES-256*.

3.3 Disponibilização das aplicações e serviços

A disponibilização da aplicação móvel foi feita através da disponibilização do pacote *Android APK (Android Package)* gerado pela compilação do projeto utilizando a ferramenta de compilação *Gradle*, para poder ser distribuído e instalado nos dispositivos *Android*.

Quanto à aplicação servidora, a mesma foi alojada numa máquina virtual *Linux* disponibilizada através da *Azure*, onde esta foi configurada para permitir ligação apenas *HTTPS* e assim como disponibilizar conexões *websocket*.

Também ficaram definidas nessa máquina como variáveis de ambiente a *connection string* para permitir ligar à base de dados *MongoDB*, assim como as credenciais de acesso para utilizar o serviço de alojamento.

Relativamente à disponibilização da base de dados, foi criada uma base de dados em um *cluster* no *MongoDB Atlas* e posteriormente corrido o script de criação das coleções. Foi necessário adicionar os *IP* referentes à máquina virtual da aplicação servidora para permitir a mesma conectar-se ao *cluster* da base de dados.

Por último, o serviço de alojamento de dados foi o serviço utilizada para o desenvolvimento da aplicação, sendo a diferença a utilização de um novo *Bucket* para armazenar novos objetos.

4. Avaliação Experimental

4.1 Testes funcionais

Para realização dos testes de avaliação experimental, a aplicação foi instalada em dois *smartphones Android*, com o âmbito de averiguar o correto funcionamento do projeto, assim como se os objetivos traçados para o mesmo foram alcançados.

Para tal, com o intuito de testar todas as situações possíveis de ocorrer na utilização da aplicação, foi decidido que no decorrer dos testes fossem criadas quatro contas de utilizador: Andreia, sexo feminino e 18 anos de idade; André, sexo masculino e 22 anos de idade; Ana, sexo feminino e 21 anos de idade; João, sexo masculino e 25 anos de idade.

Para iniciar os testes, foi instalado o APK disponibilizado e lançada a aplicação que nos levou até ao ecrã inicial, como se pode ver na figura 14 que serve apenas como página inicial da aplicação quando inicializada pela primeira.



Figura 14 – Ecrã inicial da aplicação

Posteriormente, somos reencaminhados automaticamente para o ecrã de autenticação, apresentado na figura 15.



Figura 15 – Ecrã de autenticação do utilizador

O ecrã de autenticação permite ao utilizador introduzir as suas credenciais (email e password) para se autenticar, sendo posteriormente redirecionado para o ecrã de escuta de áudios (apresentado mais à frente) se a autenticação for bem-sucedida. Caso seja a primeira vez, necessitamos de criar uma conta de utilizador primeiro, podendo fazê-lo ao clicar no ‘*Create Account*’ onde somos redirecionados para o ecrã de criação de conta de utilizador, ilustrado na figura 16.

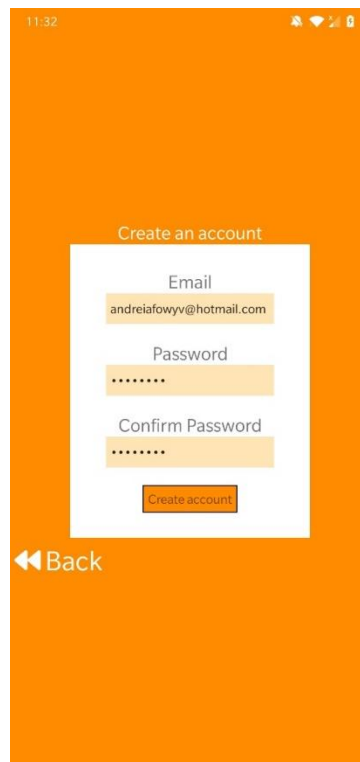


Figura 16 – Ecrã de criação de conta do utilizador

No ecrã de criação de conta de utilizador, é possível criar uma conta simplesmente introduzindo um email válido e definindo uma password que ficará associada à nova conta, esta deverá ser introduzida duas vezes para confirmação. Para a primeira conta decidimos criar o utilizador Andreia. Após definir as credenciais e submetermos para a criação da nova conta, é apresentado o ecrã para definir o perfil associado à nova conta, como se pode ver na figura 17.

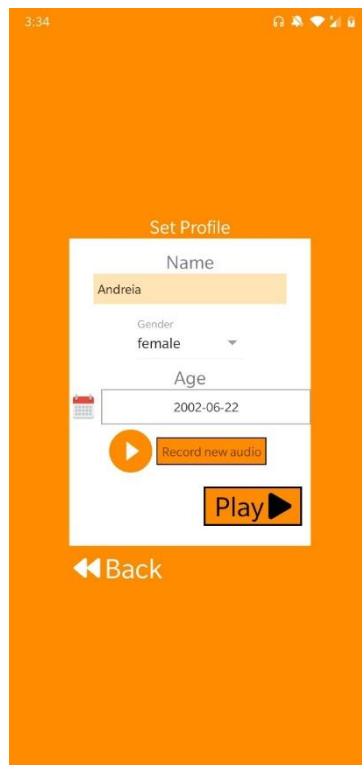


Figura 17 – Ecrã de definição do perfil do utilizador

Neste ecrã é nos dada a possibilidade de introduzir os dados que ficaram no perfil associado ao utilizador, nomeadamente o nome visível para os outros utilizadores e a sua data de nascimento, assim como o seu género. Também neste momento o utilizador deverá gravar o seu áudio pessoal, clicando no botão “*Record Audio*”, que irá mostrar um elemento modal, ilustrado na figura 18.

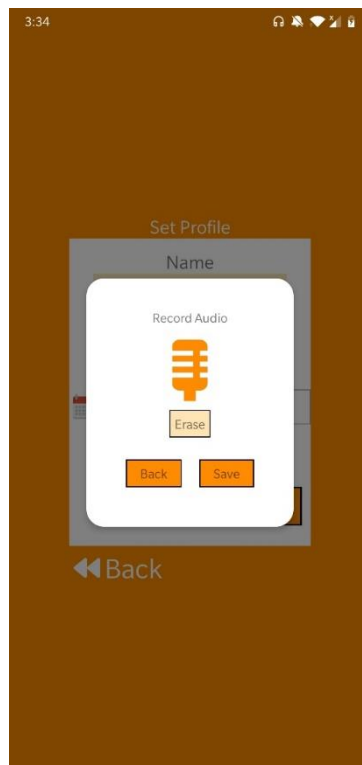


Figura 18 – Ecrã com elemento modal para gravação de áudio

É possível gravar iniciar a gravação do áudio clicando no *icon* do microfone, que mostra seguidamente um *icon* de um quadrado para parar a gravação. Após recolhida a mesma, podemos finalizar a recolha do áudio clicando no botão “*save*” e retornar para o ecrã de definição do perfil de utilizador. Tendo todos os pontos recolhidos do utilizador Andreia, é submetido o seu perfil. Após a submissão com sucesso, é nos apresentado o ecrã de escuta de áudios, representado na figura 19.

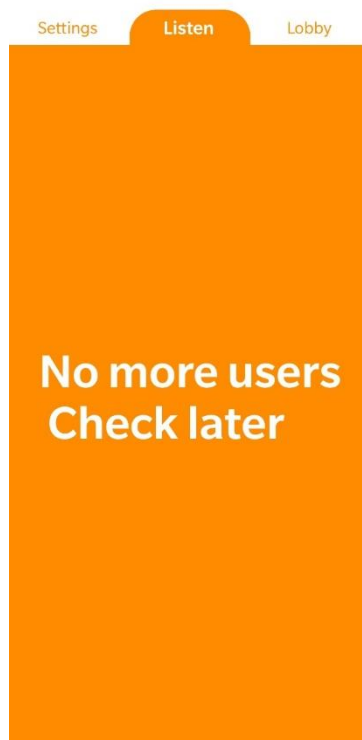


Figura 19 – Ecrã de pesquisa de utilizadores

Será para este ecrã que um utilizador autenticado é redirecionado, onde, se existirem outros utilizadores tendo em conta as configurações de pesquisa do utilizador, é dada a possibilidade de escutar os seus áudios clicando no *icon* de *play* que é demonstrado, podendo posteriormente deslizar para a esquerda para poder ouvir um novo áudio de outro utilizador descartando o que se encontra exposto no ecrã, ou para a direita para referir que gostou do áudio e pretende estabelecer contato. Quando os perfis enquadrados nas preferências do utilizador terminarem, é nos mostrada uma mensagem a referir que não existem mais resultados na pesquisa de novos utilizadores.

De referir que também é apresentada a barra de navegação superior para que o utilizador possa navegar a partir deste ecrã para o ecrã de correspondências e para ecrã de configurações, comum a estes três ecrãs.

Ao clicar nos na opção “*Settings*” da barra de navegação somos direcionados para o ecrã das configurações (figura 20), onde podemos alterar as configurações de pesquisa de novos utilizadores, nomeadamente a idade ou o género dos mesmos. Como nenhuma foi alterada, as configurações mantêm-se as *default*.

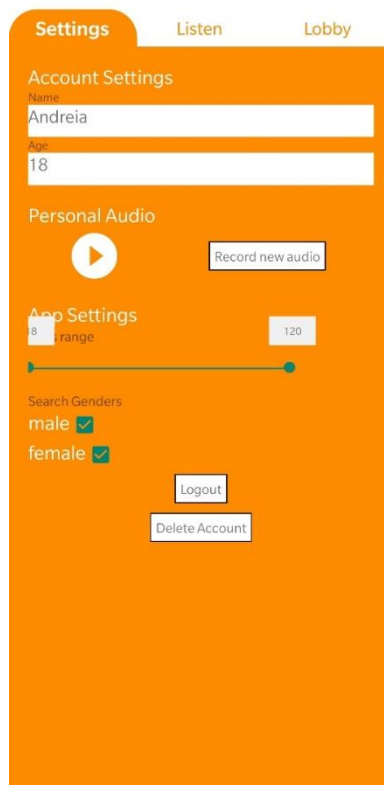


Figura 20 – Ecrã de configurações

Navegando também através da barra de navegação superior para o ecrã das correspondências, é possível verificar se o nosso utilizador tem alguma correspondência.

Como ainda não tivemos a oportunidade de ter uma correspondência, não é possível encontrar alguma, como se pode ver na figura 21.



Figura 21 – Ecrã de correspondências

Com o outro *smartphone*, executámos o mesmo processo de criação da conta para os outros três utilizadores. No processo de criação do último utilizador que faltava criar, o utilizador André, após ser submetido o seu perfil e ser redirecionado para o ecrã de escuta de áudios, podemos verificar que existe um resultado da pesquisa de novos utilizadores, sendo ele o utilizador Andreia, como demonstrado na figura, onde podemos simplesmente clicar no ícone de *play* e ouvir o áudio associado ao utilizador, visível na figura 22.

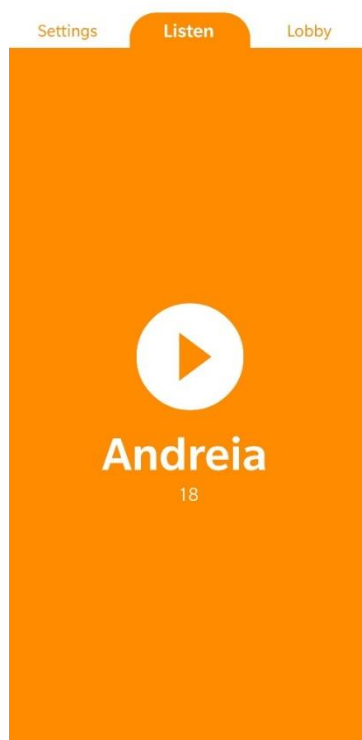


Figura 22 – Ecrã de pesquisa de utilizador, com um resultado

Seguidamente, utilizando a conta do utilizador André, decidimos ir ao menu dos *Settings* e alterar as configurações de pesquisa, de forma a pesquisar apenas utilizadores do género feminino, com idades compreendidas entre os 20-25 anos, ilustrado na figura 23.

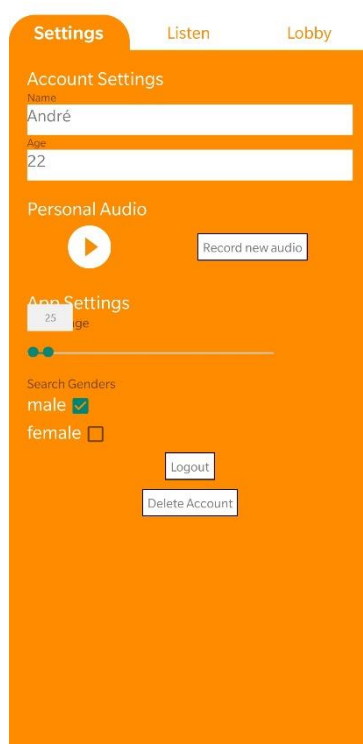


Figura 23 – Ecrã de configurações, com configurações de pesquisa alteradas

Posteriormente, voltámos ao ecrã de procura e apareceu como primeiro resultado o utilizador Ana, decidimos ouvir o áudio e aprovar, podendo visualizar se essa mesma ação na figura 24.



Figura 24 – Ecrã de pesquisa de utilizadores, ilustrando a ação de aprovação de um utilizador

Após darmos gosto, foi no apresentado no ecrã a referência que já não existiam mais resultados na procura de novos utilizadores, tendo em conta as configurações de pesquisa atuais.

Decidimos então fazer *logout* na conta do utilizador André e entrar com a conta do utilizador João, tentando entrar com uma password diferente daquela que introduzimos para a criação de conta, ao qual foi nos mostrado uma mensagem de erro ao autenticar ilustrada na figura 25.



Figura 25 – Ecrã de login, ilustrando um erro de autenticação

Depois de introduzirmos as credenciais corretas do utilizador e deste ser autenticado somos redirecionados para o ecrã dos áudios, podemos verificar os áudios dos restantes utilizadores e inclusive dar gosto no utilizador Andreia.

Voltando ao *smartphone* com a conta do utilizador Andreia iniciada, decidimos alterar o áudio associado ao utilizador indo ao ecrã dos *Settings* e clicar para gravar um novo áudio apresentado na figura 26.

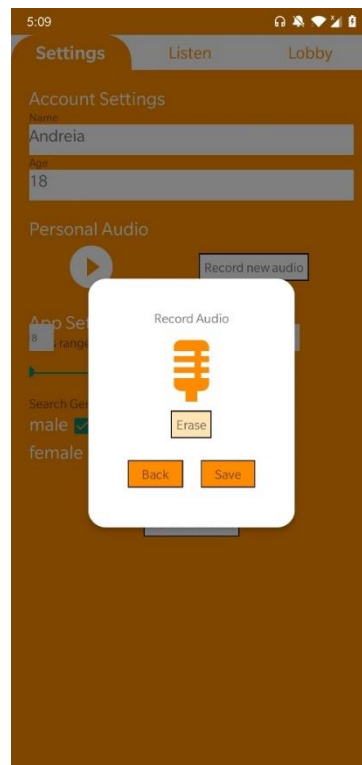


Figura 26 – Ecrã de configurações, ilustrando a gravação de um novo áudio pessoal

Também decidimos alterar as configurações de pesquisa para apenas utilizadores do sexo masculino dentro da faixa etária dos 20 aos 25 e voltar para o ecrã da pesquisa de utilizadores, onde apareceu como resultado da pesquisa o utilizador João, ao qual ouvimos o áudio e não demos gosto e posteriormente aparecer o utilizador André, ao qual ouvimos o áudio e demos gosto.

Com o smartphone onde se encontrava o utilizador João, foi feito *logout* da conta do utilizador e entramos com a conta do utilizador André e decidimos alterar as configurações, alterámos de preferência por género feminino e dentro da faixa etária dos 20 aos 25 para procurar novamente pelos dois géneros e sem especificar margem de idades.

Posteriormente voltando novamente para o ecrã de procura, podemos voltar a novamente a ver nos resultados de pesquisa o utilizador Andreia.

Clicando sobre o áudio do utilizador Andreia, podemos ouvir o novo áudio associado à sua conta e seguidamente decidimos dar gosto, aparecendo uma caixa de texto com uma mensagem a informar de tínhamos tido correspondência mútua e é nos dada a possibilidade de iniciar uma interação com o utilizador ou continuar à procura de outros utilizadores, como se pode ver na figura 27.

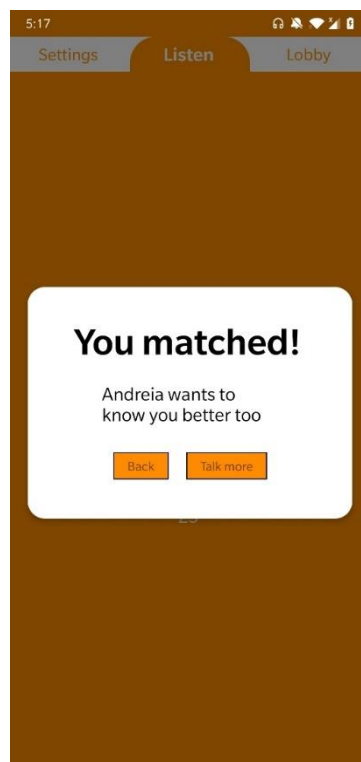


Figura 27 – Ecrã das pesquisas, mostrando uma mensagem de correspondência

Clicando na opção *Back*, voltamos aos resultados e verificamos no resultado da pesquisa o utilizador João, ao qual não demos gosto ficando o ecrã de sem novos resultados.

Posteriormente vamos para o lobby de matches, onde podemos encontrar o utilizador com que tivemos correspondência, demonstrado na figura 28.



Figura 28 – Ecrã de correspondências

Ao clicar na nova correspondência, o ecrã de chat é *renderizado*. Neste ecrã, é nos dada a possibilidade de interagir inicialmente apenas através de mensagens de áudio. Iniciamos a interação ao gravar uma mensagem de voz, apresentado na figura 29.



Figura 29 – Ecrã de chat, com uma mensagem de áudio

No outro *smartphone*, podemos verificar a receção da mensagem de áudio e posteriormente ouvi-la. Seguidamente gravamos também uma mensagem de áudio e enviamos, como possível de visualizar na figura 30.



Figura 30 – Ecrã de chat, com interação inicial áudio entre dois utilizadores

Após o envio de três mensagens de áudio de cada utilizador, é desbloqueado o envio de mensagens de texto. Podemos então testar em ambos o envio de mensagens de texto, como apresentado na figura 31.

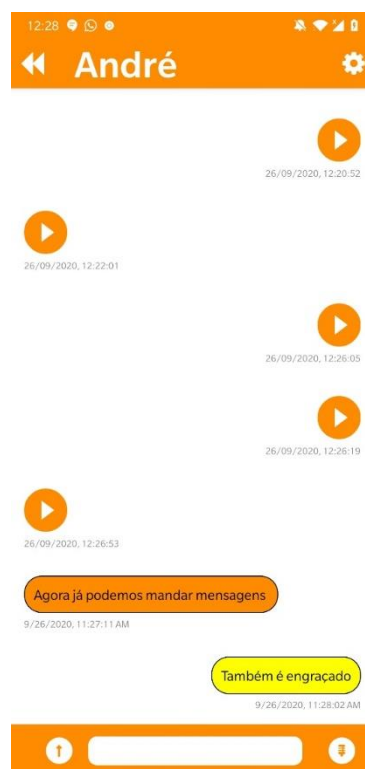


Figura 31 – Ecrã de chat, com interação de áudio e de texto

Posteriormente, através do *ícone* no canto superior direito, ter a opção de eliminar a correspondência, visível na figura 32.

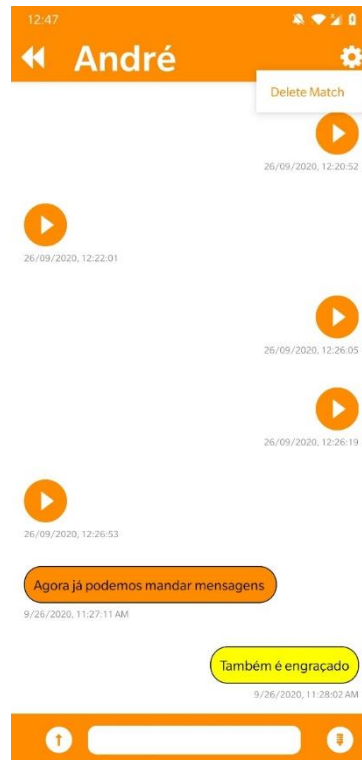


Figura 32 – Ecrã de chat, ilustrando a opção de apagar a correspondência

Ao clicar nele, a correspondência é apagada e somos redirecionados para o ecrã de matches, onde podemos verificar que a mesma desapareceu, visível na figura 33.



Figura 33 – Ecrã de correspondências, sem nenhuma correspondência

4.2 Análise dos resultados obtidos

No decorrer dos testes realizados na avaliação experimental da aplicação, foram obtidos constantemente os valores da base de dados para analisar e verificar os resultados obtidos, diferenciando em cinco temas: criação dos utilizadores, interação na escolha de utilizadores, alteração das configurações de pesquisa, correspondência entre utilizadores e eliminação de correspondência.

Após a criação dos utilizadores, podemos verificar que foram criados os documentos na coleção *UsersDetails* referentes a cada utilizador, verificando na figura 33 os documentos dos utilizadores André e Andreia.

```
{
  "_id": {
    "$oid": "5f6eb6f3cee02d00445512b5"
  },
  "email": "andrefowyv@hotmail.com",
  "name": "André",
  "gender": "male",
  "age": "1998-06-22",
  "audioFile": "950a913a-e89b-46a1-abd1-630db0b5bc6e.aac"
}

{
  "_id": {
    "$oid": "5f6ea8decee02d00445512a9"
  },
  "email": "andreiafowyv@hotmail.com",
  "name": "Andreia",
  "gender": "female",
  "age": "2002-06-22",
  "audioFile": "3127722a-04c9-40b3-bd0d-1fddf103cc60.aac"
}
```

Figura 33 – Dados armazenados na coleção *UsersDetails* dos utilizadores André e Andreia

Podemos verificar os dados introduzidos do nome, género e data de nascimento, assim como o nome do ficheiro do áudio pessoal associado ao utilizador que foi armazenado no serviço de alojamento de dados.

As configurações de pesquisa puderam ser analisadas através da coleção *UsersSearchSettings*, onde analisamos o documento relativo às configurações do utilizador Andreia que foram alteradas no decorrer dos testes, podendo ver na figura 34 as configurações iniciais.

```
{
  "_id": {
    "$oid": "5f6ea80ccee02d00445512a8"
  },
  "email": "andreiafowyv@hotmail.com",
  "minSearchAge": 18,
  "maxSearchAge": 100,
  "searchGenders": ["male", "female"]
}
```

Figura 34 – Dados armazenados na coleção *UsersSearchSettings* das configurações do utilizador Andreia

Seguidamente às alterações, foi analisado o mesmo documento e verificar as alterações efetuadas para a pesquisa de utilizadores apenas do género feminino e com idade compreendida entre os 20 e 25 anos, visíveis na figura 35.

```
{
  "_id": {
    "$oid": "5f6ea80ccee02d00445512a8"
  },
  "email": "andreiafowyv@hotmail.com",
  "minSearchAge": 20,
  "maxSearchAge": 25,
  "searchGenders": ["male"]
}
```

Figura 35 – Dados armazenados na coleção UsersSearchSettings das configurações do utilizador Andreia, após terem sido alteradas

Após as interações de escolha de utilizadores, podemos analisar os perfis eleitos de cada utilizador através da coleção *UsersChoises* e podemos verificar as interações relativas a cada utilizador, onde podemos ver os perfis aprovados e não aprovados de cada utilizador, possível de visualizar na figura 35 as interações do utilizador Andreia.

```
{
  "_id": {
    "$oid": "5f6ea80ccee02d00445512a7"
  },
  "email": "andreiafowyv@hotmail.com",
  "likedUsers": [{
    "email": "andrefowyv@hotmail.com"
  }],
  "dislikedUsers": [{
    "email": "joaofowyv@hotmail.com"
  }]
}
```

Figura 35 – Dados armazenados na coleção UsersChoises referentes as interações do utilizador Andreia

Também analisámos a coleção *UsersMatches*, onde foi possível analisar a correspondência entre o utilizador André e Andreia, tendo acesso às mensagens armazenadas, onde inclusive verificamos que as primeiras mensagens foram apenas de áudio através da figura 36.

```

{
  "_id": {
    "$oid": "5f6f22ab3ac4cb00459a2e88"
  },
  "emails": ["andreiafowyv@hotmail.com", "andrefowyv@hotmail.com"],
  "messages": [
    {
      "id": "d65e7f31-c622-4c3c-8ae5-d574b3e8d5a9",
      "user": "andrefowyv@hotmail.com",
      "type": "AUDIO",
      "date": "26/09/2020, 12:18:54",
      "content": "b2d854a2-6d81-4895-b40e-1834e332a459.aac",
      "state": "sended"
    },
    {
      "id": "0e639da4-c981-446a-98b1-d3ad811ba0a9",
      "user": "andreiafowyv@hotmail.com",
      "type": "AUDIO",
      "date": "26/09/2020, 12:20:28",
      "content": "fb2a277f-946c-4b6f-aae2-0ba47e182d04.aac",
      "state": "sended"
    }
  ]
}

```

Figura 35 – Dados armazenados na coleção UsersMatches referentes as interações existentes na correspondência entre os utilizadores André e Andreia.

Por fim, após a eliminação do match, não foi possível observar novamente na coleção *UsersMatches* a correspondência entre ambos, apesar de a interação entre utilizadores estar referida nas escolhas de cada utilizador, o que fará com que os utilizadores não possam voltar a ter correspondência novamente através da pesquisa de novos utilizadores.

Com os resultados obtidos, podemos verificar que a criação de utilizadores funcionou como era espectável. Assim como na pesquisa e interação entre utilizadores, podemos verificar que a alteração das configurações de pesquisa altera o resultado de pesquisa de novos utilizadores e depois de aparecer um utilizador no ecrã, sendo ou não aprovado, esse perfil não voltará a aparecer como resultado de pesquisa.

Quanto à correspondência entre utilizadores, verificámos que é possível apenas interagir através do envio de mensagens de áudio, podendo gravar e escutar os áudios sem que haja conflito ou descarregamento do ficheiro áudio incorreto, sendo posteriormente disponibilizada a funcionalidade de envio de mensagens de texto.

A eliminação da correspondência também comprovou que a interação entre utilizadores deixa de ser possível, quer por a própria correspondência deixar de aparecer nos resultados de correspondências dos utilizadores, quer por os mesmos não aparecerem mais nos resultados de pesquisa de novos utilizadores um do outro.

5. Conclusões

Neste trabalho foi explorada a ideia de como a interação virtual é cada vez mais utilizada para conhecer novas pessoas senão a forma mais utilizada para esse fim nos dias de hoje. Houve a possibilidade de analisar e encontrar uma forma de tornar a experiência virtual mais pessoal e dinâmica, verificando que existem outras alternativas para descobrir e conectar com outras pessoas.

A solução desenhada permitiu satisfazer os objetivos delimitados na fase inicial do projeto, endereçando os requisitos que foram levantados na projeção do mesmo. Foi obtida uma solução facilmente escalável que permite a comunicação entre utilizadores quase em tempo real que poderia no futuro ser facilmente disponibilizada para o sistema operativo *iOS* de forma a permitir a utilização da mesma não só em dispositivos *Android*.

Contudo, apesar de ter sido traçado um plano inicial de trabalho com objetivos funcionais concretos definidos, foi verificada uma falha no planeamento técnico, visto não ter sido definido desde logo um plano estruturado do funcionamento completo da solução, o que originou a que durante o decorrer do projeto fossem levantadas questões e dúvidas que deveriam ter sido endereçadas no início, erro esse que levou a que existissem atrasos pela indefinição das mesmas.

Além disso, existem outros pontos que poderiam ser melhorados, nomeadamente a nível da segurança dos dados dos utilizadores da aplicação. Devia ter sido feita a encriptação dos dados pessoais e das mensagens por conterem informação confidencial dos utilizadores e a aplicação deveria não permitir a obtenção de ficheiros de áudio por qualquer utilizador autenticado através da aplicação servidora, implementando uma lógica que permitisse apenas obter os áudios das conversas em que o utilizador estivesse inserido ou os áudios pessoais que definem os utilizadores.

Outro ponto a ser revisto seria a integração dos serviços da aplicação móvel com a aplicação servidora, nomeadamente a nível do uso dos *websockets*, visto que estes poderiam ser implementados de forma a permitir um maior controlo do estado atual da aplicação móvel, com confirmação da receção dos eventos emitidos por ambas as partes e permitindo obter maior certeza de que a mesma contenha os dados mais atualizados.

Para além das melhorias anteriormente faladas, o projeto tem bastante espaço para implementação de novas funcionalidades que permitam tornar a aplicação mais apelativa e com mais características para além da interação limitada a duas pessoas. Uma funcionalidade poderia ser a implementação de pesquisa de utilizadores tendo em conta temas em que ambos tivessem interesse ou permitir a interação conjunta de um grupo de utilizadores que partilhasse um tema em comum e que o quisesse discutir.

A realização deste projeto permitiu utilizar tecnologias e *frameworks* que têm tido cada vez mais adesão e impacto atualmente, sendo o conhecimento que adquiri em todo este processo um contributo para o meu futuro profissional.

Referências

- [1] “Tinder - Corresponda. Converse. Encontre-se.,” IAC, 14 03 2020. [Online]. Available: <https://tinder.com/>.
- [2] “Happn - Encontre as pessoas que cruzam o seu caminho,” HAPPN, 14 03 2020. [Online]. Available: <https://www.happn.com/pt-br/>.
- [3] “Badoo - Conheça novas pessoas no Badoo, Faz novas amizades ...,” Badoo Trading Limited, 14 03 2020. [Online]. Available: <https://badoo.com/pt-pt/>.
- [4] “React Native,” Facebook Inc., 17 03 2020. [Online]. Available: <https://reactnative.dev/>.
- [5] “Android,” Google LLC, 14 03 2020. [Online]. Available: https://www.android.com/intl/pt_pt/.
- [6] “iOS,” Apple Inc., 14 03 2020. [Online]. Available: <https://www.apple.com/pt/ios/ios-13/>.
- [7] “Node.js,” Joyent, Inc., 17 03 2020. [Online]. Available: <https://nodejs.org/en/>.
- [8] “MongoDB,” MongoDB, Inc., 30 03 2020. [Online]. Available: <https://www.mongodb.com/>.
- [9] “Amazon Simple Storage Service (Amazon S3),” Amazon Web Services, Inc, 20 04 2020. [Online]. Available: <https://aws.amazon.com/pt/s3/>.

A.1 Wireframes

