UNIVERSIDADE AUTÓNOMA DE LISBOA
LUÍS DE CAMÕES

DEPARTAMENTO DE ENGENHARIAS E CIÊNCIAS DA COMPUTAÇÃO
LICENCIATURA EM INFORMÁTICA DE GESTÃO
LICENCIATURA EM ENGENHARIA INFORMÁTICA

Sistema de Autenticação Digital Anti *Deepfakes*

Laboratório de Projeto

Autores: Edgar Casimiro, Miguel Fernandes, Pedro Brito, Tiago Mateus
Docente Orientador: Professor Dr. Héctor Dave Orrillo Ascama
Número dos candidatos: 19970423, 30008210, 30008361, 30010863

Julho de 2024
Lisboa

Agradecimentos

É com grande satisfação que concluímos este capítulo da nossa jornada académica. Gostaríamos de expressar a nossa mais profunda gratidão ao nosso orientador Professor Dr. Héctor Dave Orrillo Ascama, cuja orientação e sabedoria foram fundamentais para o desenvolvimento e sucesso deste trabalho. Aos meus colegas, agradecemos pela camaradagem e pelo apoio mútuo que nos permitiu superar todos os desafios. Por fim, mas não menos importante, um agradecimento especial à família, que merece uma menção especial pelo suporte inabalável e pela confiança depositada, elementos essenciais que permitiram alcançar este marco significativo. Este projeto é um testemunho do poder da dedicação coletiva e do suporte mútuo, e é com um sentimento de realização que partilhamos o com todos vós com um coração cheio de apreço.

O nosso sucesso é o reflexo da determinação de cada um de nós.

Obrigado, equipa!

Epígrafe

"A verdade raramente é pura e nunca é simples."

Autor: Oscar Wilde

Resumo

Como objetivo temos de desenvolver um sistema de autenticação que utilize biometria de voz e facial para combater a manipulação de conteúdo digital conhecida como *Deepfake.*

Na metodologia usada o projeto será apoiado pelo orientador, Prof. Dr. Héctor Dave Orrillo Ascama, e baseia-se em pesquisa de artigos, teses e pesquisa extensiva na internet. Serão utilizadas as técnicas de *Machine Learning* e redes neurais convolucionais (CNNs) para o reconhecimento e autenticação de padrões visuais e sonoros.

Como tecnologias utilizadas no desenvolvimento e suporte usamos o *Python* como linguagem de programação de alto nível utilizada para o desenvolvimento do sistema. como

base de dados do projeto utilizamos *a Firebase* que é uma Plataforma desenvolvida pelo Google. No *front-end* usamos o *Flutter* por ser um *Framework* de código aberto desenvolvido pelo Google para a criação de aplicativos móveis, assim como outros, *Google ML Kit, TFLite, FaceNet e FaceNet 512, Dart* e uma série de bibliotecas de apoio.

O nossos desafios e soluções irão centrar-se em combater o *Deepfake* que são conteúdos de vídeo ou áudio manipulados por inteligência artificial que representam um desafio para a autenticação digital. A solução proposta envolve o uso de biometria de voz e facial para verificar a autenticidade do utilizador. Também incluímos na nossa proposta o *Spoofing,* que é a falsificação de identidade em comunicações digitais. O sistema proposto visa prevenir o s*poofing* através de métodos de verificação robustos.

Além disso, o projeto envolve o uso de *Machine Learning* e redes neurais convolucionais (CNNs) para o reconhecimento e autenticação de padrões visuais e voz. Estas tecnologias foram fundamentais para combater a manipulação de conteúdo digital conhecida como *Deepfake* e para prevenir o *spoofing* em comunicações digitais.

Como resultados esperados queremos que o sistema de autenticação seja capaz de detetar e prevenir tentativas de *Deepfake* e *spoofing,* garantindo a segurança e a integridade das comunicações digitais dos utilizadores.

Para o relatório de grupo do nosso grupo composto por quatro elementos: Edgar Casimiro – 19970423, Miguel Fernandes – 30008210, Pedro Brito – 30008361, Tiago Mateus – 30010863, iremos esforçar-nos para atender às expectativas na estrutura, pesquisa e realização deste relatório. **Palavras-chave:** Deepfake; prova de vida; Biometria de voz; Biometria facial.

Abstract

Our goal is to develop an authentication system that uses voice and facial biometrics to combat the manipulation of digital content known as Deepfake.

The project will be supported by the advisor, Prof. Dr. Héctor Dave Orrillo Ascama, and is based on research from articles, theses, and extensive internet searches. We will use

Machine Learning techniques and Convolutional Neural Networks (CNNs) for the recognition and authentication of visual and audio patterns.

For development and support, we use Python as the high-level programming language. Firebase, a platform developed by Google, is used as the project database. For the front-end, we use Flutter, an open-source framework developed by Google for creating mobile applications, along with other technologies such as Google ML Kit, TFLite, FaceNet, FaceNet 512, Dart, and various supporting libraries.

Our challenges and solutions will focus on combating Deepfakes, which are video or audio content manipulated by artificial intelligence, posing a challenge for digital authentication. The proposed solution involves using voice and facial biometrics to verify user authenticity. We also address spoofing, which is identity falsification in digital communications. The proposed system aims to prevent spoofing through robust verification methods.

Furthermore, the project involves using Machine Learning and Convolutional Neural Networks (CNNs) for the recognition and authentication of visual and voice patterns. These technologies are crucial for combating digital content manipulation known as Deepfake and preventing spoofing in digital communications.

We expect the authentication system to detect and prevent Deepfake and spoofing attempts, ensuring the security and integrity of users' digital communications.

Our group, consisting of four members: Edgar Casimiro – 19970423, Miguel Fernandes – 30008210, Pedro Brito – 30008361, Tiago Mateus – 30010863, will make every effort to meet expectations regarding the structure, research, and completion of this report.

Keywords: Deepfake; liveness detection; voice biometrics; facial biometrics.

Índice

Lista de Fotografias/Ilustrações

Lista de Siglas e Acrónimos

Deepfake	Conteúdos de vídeo ou áudio manipulados por inteligência artificial
spoofing	Falsificação de identidade em comunicações digitais
python	Linguagem de programação de alto nível
ML	*Machine Learning* - Aprendizado de Máquina
Firebase	Plataforma desenvolvida pelo Google, Base de dados
Flutter	framework de código aberto desenvolvido pelo Google, aplicativos dispositivos móveis
Insights	Perceções ou entendimentos profundos derivados de dados
Software	Programas, aplicativos e sistemas operacionais executados no hardware
Hardware	Componentes físicos de um sistema computacional
Fakenews	Informações deliberadamente enganosas ou fabricadas
Phishing	Forma de fraude online, obter acesso não autorizado a
Haar	Detetor de características de Haar é uma abordagem para reconhecimento de padrões visuais em imagens
CNNs	Rede neural artificial, aplicadas no reconhecimento de imagens e vídeos.
Deep Learning	Aprendizado profundo português, redes neurais profundas para aprender e representar dados de forma hierárquica
Pooling	Operação comumente usada em redes neurais convolucionais (CNNs)
Embeddings	Representação de dados num espaço de caraterísticas dimensão reduzida
API	*Application Programming Interface* - Interface de Programação de Aplicações
UI	*User Interface* – Interface Utilizador
GANs	Generative Adversarial Networks - Redes Generativas Adversariais
liveness	Prova de vida
MFCC	*Mel-frequency cepstral coefficients* - Coeficientes Cepstrais de Frequência Mel
Eigenfaces	Autofaces, técnica utilizada no reconhecimento facial
Back-end	Desenvolvimento responsável pelo servidor, base dados e lógica aplicacional
Front-end	Desenvolvimento visível ao utilizador, design e interatividade
Backoffice	Operações internas que não são visíveis ao utilizador
Node.js	Ambiente de execução JavaScript no servidor de alta performance

DTW	*Dynamic Time Warping* - algoritmo utilizado para medir a similaridade entre duas sequências temporais, padrões
LFCC	Coeficientes de Filtro de Frequência
Frames	Uma estrutura de dados usada para representar um conceito
AZURE	Microsoft Azure é uma plataforma de computação em nuvem

1 Introdução

1.1 Descrição do problema

Nos últimos anos, a proliferação de *deepfakes* emergiu como uma ameaça significativa e complexa que afeta diversos setores da sociedade de maneiras profundas e variadas. Este crescimento é impulsionado pelos avanços contínuos em aprendizagem de máquina (*machine learning*) e inteligência artificial, que têm possibilitado a criação de vídeos e gravações de áudio falsos cada vez mais sofisticados e convincentes.

Os *deepfakes* possuem a habilidade de manipular e distorcer gravações de áudio e vídeo, esta nova tecnologia tem a capacidade de transformar a voz e ações de alguém em narrativas enganosas representando assim uma ameaça multifacetada para indivíduos, organizações e sociedade como um todo.

O grande desenvolvimento da inteligência artificial capacitou a estas ferramentas a possibilidade de alcançar um nível de sofisticação sem igual. A fronteira entre a informação e conteúdo digital real e o que é manipulado é cada vez mais difusa, tornando assim a tarefa de distinguir a realidade do que é manipulado muito complexa.

Esta erosão da confiança no conteúdo digital tem consequências de longo alcance, colocando em risco a integridade da informação, corrompendo processos democráticos e alimentando a disseminação de desinformação.

As implicações dos *deepfakes* vão além do domínio do conteúdo digital, lançando uma sombra ameaçadora sobre nossas vidas pessoais. A capacidade de fabricar conteúdo incriminatório ou difamatório representa uma grave ameaça à privacidade e posição social dos indivíduos. A utilização de *deepfakes* para fins maliciosos, como chantagem, extorsão ou a erosão da confiança pública em figuras de autoridade, não pode ser subestimado.

Ao navegarmos por este território desconhecido, é imperativo reconhecer o impacto profundo dos *deepfakes* em nosso mundo. Devemos enfrentar os desafios impostos por esta tecnologia, não com medo e apreensão, mas com um compromisso resoluto de entender, mitigar e combater seus efeitos insidiosos. Somente através de um esforço conjunto, abrangendo

inovação tecnológica, educação e reforma de políticas, podemos proteger a integridade da informação, proteger nossas identidades pessoais e preservar as bases de nossa sociedade.

Esta introdução prepara o terreno para uma exploração abrangente dos *deepfakes*, abordando seus fundamentos técnicos, implicações sociais e potenciais contramedidas (*anti deepfakes*). Ela destaca a urgência de abordar esta ameaça emergente e enfatiza a necessidade de uma abordagem multifacetada para proteger nossas realidades digitais e físicas.

1.1.1 Crescimento da ameaça *Deepfakes*

O impacto dos *deepfakes* transcende a mera disseminação de desinformação ou o uso para entretenimento. Esta tecnologia avançada representa uma ameaça substancial e crescente à integridade e confiabilidade dos sistemas de autenticação e segurança cibernética em uso atualmente.

Com o aumento exponencial na sofisticação e disseminação dos *deepfakes*, métodos tradicionais de verificação de identidade, como senhas e sistemas biométricos, tornam-se cada vez mais obsoletos e vulneráveis.

À medida que os *deepfakes* são mais acessíveis e fáceis de criar, o número de ataques de *spoofing* que utiliza esta tecnologia está em ascensão. Isso compromete significativamente a segurança de transações online, a proteção de dados pessoais e a integridade dos sistemas organizacionais.

Além disso, a capacidade de *deepfakes* de imitar vozes e aparências de maneira quase perfeita amplia o espectro de ameaças, tornando ainda mais difícil para os sistemas de autenticação distinguir entre pessoas reais e falsificações.

1.1.2 Vulnerabilidade dos sistemas de autenticação tradicionais

A vulnerabilidade dos sistemas de autenticação coloca em cheque não apenas a confiança depositada em plataformas digitais e sistemas de informação, mas também evidencia a fragilidade dos sistemas de autenticação tradicionais frente à crescente sofisticação dos

deepfakes. A capacidade dessas tecnologias de imitar vozes, expressões faciais e até mesmo comportamentos humanos de forma quase indistinguível torna métodos de verificação baseados em conhecimento pessoal, como senhas e perguntas de segurança, altamente suscetíveis a ataques de *spoofing*.

À medida que os *deepfakes* se tornam mais avançados e difundidos, a eficácia de métodos biométricos, como reconhecimento facial e de impressões digitais, também é questionada. A capacidade de *deepfakes* em reproduzir com precisão características biométricas de indivíduos reais representa uma ameaça significativa, comprometendo a integridade e a confiabilidade desses sistemas de autenticação.

Além disso, o uso malicioso de *deepfakes* por parte de atores de ameaças para se passarem por indivíduos reais ou disseminarem informações falsas tem implicações sociais e políticas profundas. Pode ir desde a manipulação de narrativas políticas e eleições até a perpetração de esquemas de fraude e extorsão. Esta capacidade de influenciar e manipular a perceção pública amplia o potencial do dano, fazendo com que a mitigação dessas ameaças seja um desafio cada vez mais complexo e urgente.

Neste contexto, torna-se imperativo não apenas reavaliar os métodos de autenticação existentes, mas também desenvolver e implementar soluções inovadoras e robustas que possam resistir aos avanços contínuos em *deepfake* e garantir a segurança e privacidade das informações sensíveis e pessoais.

1.2 Objetivos

Para a conclusão satisfatória deste trabalho é de extrema importância a definição de objetivos a alcançar. A criação de uma estratégia e definição de metas bem como tarefas e atividades a serem executadas pelos membros responsáveis deste projeto irão ser imperativas para excelente conclusão deste sistema de autenticação *anti deepfakes.*

1.2.1 Objetivo geral

O principal objetivo deste relatório é fornecer uma descrição detalhada de uma aplicação para um sistema de autenticação anti *deepfakes*.

Para que este trabalho seja realizado com sucesso, teremos de nos focar em metodologias e técnicas de tecnologia sofisticada na identificação destas ameaças tão reais nos dias correntes que são os *deepfakes*.

Com isto pretendemos então desenvolver uma aplicação que seja capaz não só de fazer a validação através de um email e password, mas também utilizando a capacidade de reconhecimento facial e biometria de voz para identificar o individuo que está a tentar aceder a esta mesma tecnologia.

Este sistema será desenvolvido com a ajuda de diversos componentes tecnológicos. Iremos utilizar uma base de dados, conforme figura 1, onde iremos manter, guardar os dados e toda a informação dos utilizadores tal como o reconhecimento facial do utilizador que será recolhido através da camara de qualquer dispositivo tecnológico e a biometria de voz que será então guardada através do microfone desses mesmos dispositivos.

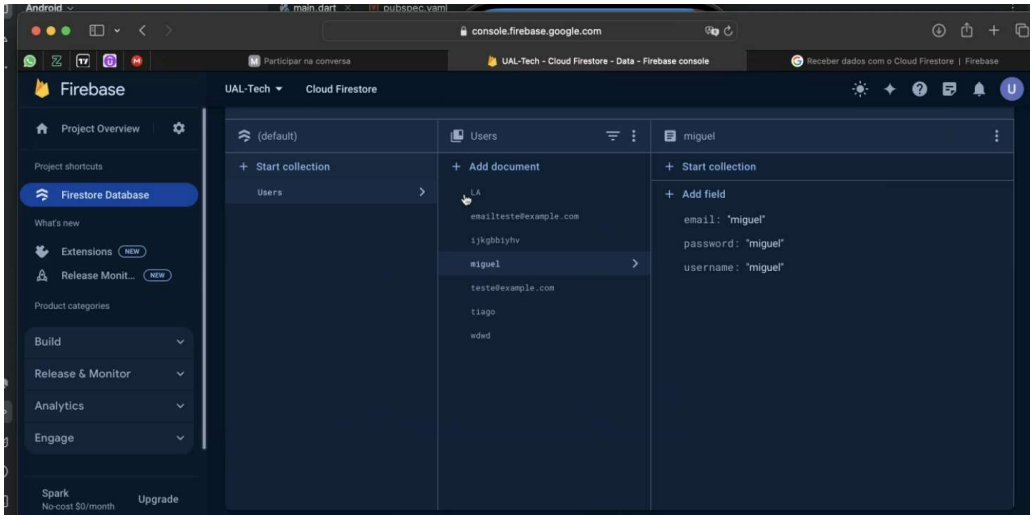

Figura 1 - Aplicativo Firebase

Fonte: Autores

Iremos utilizar um servidor desenvolvido utilizando a linguagem de programação *python* onde será feita as ligações, gestão e interação entre as restantes tecnologias tais como base de dados e interface de utilizador. Para o desenvolvimento deste servidor a utilização de bibliotecas já existentes é fundamental para implementação deste sistema. Neste servidor é também onde os dados recolhidos irão ser transferidos com a nossa base de dados que será desenvolvida com a tecnologia *Firebase* de onde tentamos aceder de forma a utilizar a metodologia já desenvolvida para esta mesma aplicação.

Teremos também uma interface de utilizador que será desenvolvida utilizando a tecnologia *Flutter*. Aqui é onde o utilizador irá fazer os seus registos, conforme demonstrado na figura 2, e terá então acesso ao sistema de autenticação anti *deepfakes* registando assim todos os seus detalhes.

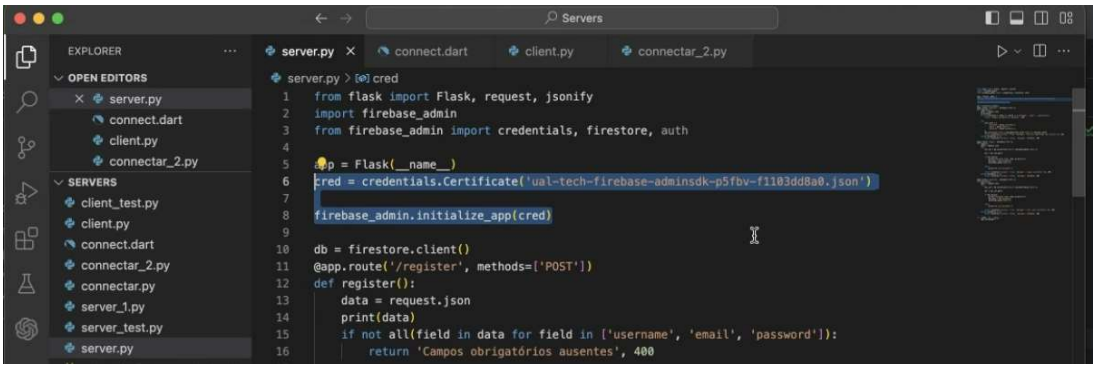

Figura 2 - Credênciais Firebase

Fonte: autores

1.2.2 Objetivos específicos

A necessidade de definição de objetivos mais específicos para o desenvolvimento desta aplicação é fundamental para que a mesma seja concluída com sucesso. Definimos aqui então alguns objetivos direcionados ao negócio e também a forma de mitigação dos mesmos.

- Risco de reputação
  - Implementação de sistema de deteção de *deepfakes* em tempo real de forma a monitorizar a autenticação dos utilizadores.
- Segurança

- Desenvolvimento de medidas de segurança avançadas para detetar e prevenir ataques e esquemas de engenharia social que utilizam tecnologias *deepfake.*

- Conformidades legais
  - Aprimorar os controlos de integridade de dados de forma a fazer uma validação da autenticidade do conteúdo submetido por cada utilizador.

1.3 Justificativa

A justificativa para a realização deste sistema de autenticação anti *deepfake* reside, tal como explicado anteriormente, na grande capacidade dos *deepfakes*, de criar vídeos e gravações de áudio falsos, altamente convincentes. Isto demonstra um desafio aos métodos tradicionais de autenticação e verificação de identidade. Dado o rápido avanço e acessibilidade da tecnologia de *deepfake*, é crucial desenvolver soluções robustas de autenticação *anti deepfake* por várias razões convincentes:

- Proteção de Informações Pessoais e Sensíveis:
  - Desenvolver sistemas de autenticação *anti-deepfake* pode fornecer uma camada adicional de segurança para proteger estas informações de exploração maliciosa que são os *deepfakes*.
- Preservação da Confiança em Plataformas Digitais:
  - A confiança é o pilar das interações e transações digitais. A prevalência dos *deepfakes* reduz essa confiança ao comprometer a autenticidade e confiabilidade do conteúdo e interações digitais. Ao implementar sistemas de autenticação *anti-deepfake*, as organizações podem reforçar a confiança nas plataformas e serviços utilizados, tranquilizando assim os utilizadores de que as suas informações e interações estão seguras e autênticas.
- Prevenção de Fraudes e Crimes Virtuais:
  - Os *deepfakes* são cada vez mais utilizados para atividades fraudulentas, incluindo tentativas de imitação pessoal, fraudes financeiras e roubo de identidade. Sistemas de autenticação *anti-deepfake* podem ajudar a detetar e prevenir esses tipos de crimes virtuais, verificando a

autenticidade dos utilizadores e transações, reduzindo assim os riscos financeiros e de reputação associados a tais atividades.

- Implicações Éticas e Sociais:
  - O uso de *deepfakes* para a imitação pessoal e desinformação tem profundas implicações éticas e sociais. Ao desenvolver sistemas eficazes de autenticação anti *deepfake*, podemos contribuir para mitigar esses impactos negativos e promover um ambiente digital mais confiável e seguro para todos os indivíduos e organizações.
- Adequação às Medidas de Segurança Futuras:
  - À medida que a tecnologia de *deepfake* continua a evoluir e a tornar-se mais sofisticada, é essencial desenvolver soluções de autenticação adaptativas e futuras.

    Investir em autenticação anti *deepfake*, irá garantir que as organizações tenham uma melhor preparação para enfrentar ameaças emergentes e manter medidas de segurança robustas diante de paisagens tecnológicas em constante evolução.

Em suma, o desenvolvimento de autenticação de sistema *anti deepfake* não é apenas importante, mas também imperativo na era digital de hoje. Ao abordar as vulnerabilidades apresentadas pela tecnologia de *deepfake*, podemos melhorar a segurança, confiabilidade e integridade de plataformas digitais, proteger informações pessoais e sensíveis e mitigar os riscos de fraudes e crimes cibernéticos. Além disso, investir em sistemas de autenticação *anti deepfake* está alinhado com considerações éticas e responsabilidades sociais, contribuindo para um futuro digital mais seguro para todos.

1.4 Estrutura do trabalho

O presente relatório encontra-se então estruturado da seguinte forma:

- Introdução
  - Esta secção é constituída pela introdução ao tema pedido, é feita uma breve explicação ao tópico descrevendo qual o problema atual com referência aos ataques de *deepfakes* e com enfase na vulnerabilidade dos

sistemas de autenticação atuais. São descritos os objetivos gerais e específicos necessários para o desenvolvimento deste sistema e também uma breve explicação acerca da necessidade de melhoria continua a estes sistemas de autenticação.

- Fundamentação teórica
    - Irá ser feita uma descrição dos diversos tópicos mais abrangentes para as soluções anti *deepfakes*, tais como:
        - *Deepfakes:* Conceito, técnicas e ameaças;
        - Reconhecimento facial: Conceito e técnicas;
        - Biometria de voz: Fundamentos e aplicações;
        - Redes Neurais e aprendizagem profunda.

Todos os tópicos acima mencionados serão a base para o desenvolvimento do sistema de autenticação anti *deepfakes*, serão vastamente descritos após uma extensa pesquisa de informação acerca dos mesmos.

- Sistema de autenticação anti *deepfakes*
    - Neste capítulo iremos demonstrar como procedemos com a implementação do caso de uso proposto. Trataremos da representação da arquitetura do sistema através de diagramas, demonstraremos quais as ferramentas utilizadas (*hardware* e *software*), metodologias, procedimentos, bibliotecas e métodos de implementação utilizados para o desenvolvimento do sistema de autenticação anti *deepfakes*. Iremos também abordar cada modulo em detalhe (reconhecimento facial, biometria de voz e modulo de fusão) explicando técnicas e conceitos utilizados para implementação deste sistema.
- Implementação e experimentação
    - Neste capítulo, é detalhado o processo de implementação prática de um sistema de autenticação anti *deepfakes*. Explora as tecnologias e algoritmos utilizados na criação do sistema, bem como os desafios e considerações durante a fase de experimentação para garantir a eficácia e robustez do sistema.
- Testes e avaliação

- O foco deste capítulo é a apresentação dos testes realizados para avaliar a eficácia do sistema de autenticação anti *deepfakes*. Discute os critérios de avaliação, metodologias de teste e os resultados obtidos, fornecendo uma análise crítica da performance e segurança do sistema em diferentes cenários e condições.

- Conclusões
  - Neste capítulo final, são apresentadas as conclusões principais derivadas da implementação, experimentação, testes e avaliação do sistema de autenticação anti *deepfakes*. Resume os principais *insights*, descobertas e implicações práticas, bem como recomendações para futuras pesquisas e desenvolvimentos na área.
- Bibliografia
  - A seção de bibliografia lista as fontes de referência utilizadas ao longo do trabalho, incluindo artigos académicos, livros, documentos técnicos e outras publicações relevantes. Organizada de acordo com as normas bibliográficas adequadas, esta seção valida e sustenta as informações e argumentos apresentados ao longo do trabalho.
- Anexos
  - Os anexos são seções opcionais de um trabalho académico onde são incluídas informações complementares que são relevantes para o entendimento do conteúdo principal, mas que não são essenciais para o desenvolvimento do argumento central do trabalho. Eles podem incluir dados brutos, tabelas extensas, gráficos, questionários, imagens, entre outros elementos que apoiam ou ilustram o texto principal. A inclusão de anexos permite ao autor apresentar informações adicionais sem interromper o fluxo do texto principal, proporcionando uma compreensão mais completa e detalhada do tema abordado.

2 Fundamentação Teórica

A fundamentação teórica é uma parte crucial de qualquer trabalho académico ou de investigação. Consiste na revisão e análise crítica das teorias, conceitos e estudos relevantes ao tema em questão, servindo assim como uma base sólida para o desenvolvimento deste estudo, fornecendo um enquadramento teórico que nos irá ajudar a compreender o problema investigado, delineando hipóteses, interpretando resultados e a contextualizar as descobertas dentro do campo de estudo. Em suma, a fundamentação teórica fornece a base intelectual necessária para sustentar e dar credibilidade ao trabalho por nós realizado.

2.1 *Deepfakes*: Conceito, Técnicas e Ameaças

Os *deepfakes* são uma forma de manipulação digital que utiliza técnicas avançadas de aprendizagem profunda, especificamente redes neurais convolucionais, para criar vídeos, imagens ou áudios falsos que parecem autênticos.

2.1.1 Tipos de *Deepfakes*

Existem diferentes tipos de *deepfakes*, e cada um com suas próprias características:

- Vídeo *deepfakes*: São os mais comuns e conhecidos, nos quais o rosto de uma pessoa é sobreposto ao de outra em vídeos. Isso pode ser usado para criar vídeos de indivíduos realizando ações ou falando coisas que nunca fizeram;
- Imagem *deepfakes*: Similar aos vídeos, mas envolve a sobreposição de rostos em imagens estáticas. Isso pode ser utilizado para criar fotografias falsas de pessoas em situações comprometedoras ou falsificar identidades;
- Áudio *deepfakes*: Utiliza técnicas de síntese de voz para criar gravações de áudio falsas, imitando a voz de uma pessoa. Esses áudios podem ser usados para difamar alguém ou criar falsas declarações.

2.1.2 Técnicas de geração de *Deepfakes*

As técnicas de geração de *deepfakes* evoluíram rapidamente, permitindo a criação de conteúdo cada vez mais realista. As principais técnicas incluem:

- Redes Generativas Adversariais (GANs): São modelos de aprendizagem profunda compostos por duas redes neurais, uma geradora e outra discriminadora, que competem entre si. A rede geradora cria amostras falsas, enquanto a discriminadora tenta distinguir entre as amostras reais e falsas. Esse ciclo de feedback permite às GANs resultados cada vez mais convincentes;
- Transferência de Estilo: Esta técnica aplica o estilo de uma imagem de origem a uma imagem de destino, permitindo a transferência de características visuais específicas, como expressões faciais, para uma nova imagem, mantendo expressões e movimentos naturais.

2.1.3 Impactos e riscos da tecnologia *Deepfake*

Os *deepfakes* apresentam diversos impactos e riscos, sejam sociais ou tecnológicos:

- Desinformação: Mais conhecida como *fakenews*, a disseminação deste tipo de *deepfakes* pode levar à propagação de informações falsas e desinformação, prejudicando a confiança pública e distorcendo a perceção da realidade;
- Manipulação política: Neste caso, os *deepfakes* podem ser usados para manipular vídeos de políticos ou figuras públicas, disseminando declarações falsas ou comprometedoras, o que pode afetar campanhas políticas e processos democráticos;
- Dano à reputação: Indivíduos que podem ser alvos de *deepfakes* que os retratam em situações comprometedoras ou disseminando mensagens falsas, causando danos à sua reputação e bem-estar psicológico;
- Fraudes: *Deepfakes* usados em esquemas de fraude, como falsificação de identidade em transações financeiras ou ataques de *phishing*, aumentando os riscos de crimes cibernéticos;
- Privacidade: A capacidade de criar vídeos falsos com facilidade levanta preocupações sobre a privacidade das pessoas, pois qualquer um pode ser alvo de manipulação digital sem o seu consentimento.

Em suma, os *deepfakes* representam uma ameaça significativa, exigindo uma abordagem multidisciplinar tanto a nível de regulamentação de tecnologias como no desenvolvimento de métodos de deteção e consciencialização pública sobre os seus riscos.

Exemplos:

- *Deepfake* de vídeos políticos: Durante as campanhas as eleitorais, podem surgir vídeos *deepfake* de políticos a fazer falsas declarações ou comprometedoras, com o objetivo de influenciar a opinião pública ou desacreditar candidatos;
- *Deepfake* de celebridades em vídeos pornográficos: Este é um exemplo extremamente preocupante, no qual os rostos de celebridades são sobrepostos em vídeos pornográficos, criando falsas cenas de intimidade que podem ser usadas para difamar ou chantagear as vítimas.
- *Deepfake* em vídeos de propaganda: Empresas podem criar vídeos *deepfake* para publicidade, sobrepondo os rostos de celebridades ou influenciadores em comerciais para promover produtos de forma enganosa;
- *Deepfake* em vídeos de entretenimento: Os *deepfakes* também são usados para criar paródias ou vídeos de entretenimento, como colocar o rosto de atores em cenas de filmes famosos, criando conteúdo viral na internet;
- *Deepfake* em áudios de chamadas telefónicas: Áudios *deepfake* podem ser utilizados em esquemas de fraude, como simular a voz de uma autoridade para solicitar informações confidenciais ou realizar transações financeiras fraudulentas.

No setor bancário:

No setor bancário, a complexidade das operações financeiras e o manuseio de grandes volumes de capital atraem bastantes riscos. Devido a isso, a implementação de soluções de segurança robustas torna-se crucial para prevenir:

- Fraude de identidade: Os *deepfakes* podem ser usados para criar vídeos ou áudios falsos de clientes, tentando simular uma pessoa autorizada a ter acesso a uma conta ilegalmente ou realizar transações fraudulentas;
- Ataques de *phishing* avançados: Os criminosos podem usar *deepfakes* para criar mensagens de vídeo ou áudio convincentes que parecem ser de instituições

financeiras legítimas, enganando os clientes a fornecerem informações confidenciais, como senhas ou números de conta, em ataques de *phishing*;

- Falsificação de identificação para empréstimos: Os *deepfakes* podem ser usados para falsificar documentos de identificação de clientes permitindo assim que os criminosos obtenham empréstimos fraudulentos em nome de terceiros, lesando desta forma clientes e instituições financeiras;
- Manipulação de dados financeiros: *Deepfakes* utilizados para manipular vídeos ou áudios de relatórios financeiros, criando falsas evidências de desempenho financeiro ou manipulando informações para enganar investidores ou reguladores;
- Ataques a sistemas de autenticação biométrica: Alguns sistemas bancários utilizam tecnologias biométricas, como reconhecimento facial ou de voz, para autenticação de clientes. *Deepfakes* podem ser usados para burlar esses sistemas, fornecendo imagens ou gravações falsas que se passam pelo cliente legítimo;

Todos estes exemplos ilustram e demonstram a diversidade de uso assim como os potenciais impactos negativos dos *deepfakes* nos diferentes contextos, desde política e entretenimento até questões de privacidade e segurança. No setor bancário, é representada uma ameaça à segurança dos dados e à confiança dos clientes, mostrando o quanto é importante que implementem medidas adequadas de segurança cibernética e prevenção de fraudes de forma a mitigar essas ameaças.

2.2 Reconhecimento Facial: Conceitos e Técnicas

A fundamentação teórica do reconhecimento facial oferece uma compreensão mais aprofundada dos conceitos e técnicas envolvidos, destacando a importância da biometria facial, da prova de vida e dos algoritmos de reconhecimento utilizados nesse contexto.

2.2.1 Biometria facial

A biometria facial é um ramo tecnológico que se concentra na identificação e autenticação com base nas características únicas do rosto de um indivíduo para identificação ou verificação de sua identidade. Baseia-se na análise de padrões faciais, como formato do

rosto, disposição e distância entre os olhos, tamanho do nariz, boca, entre outros, para criar uma representação digital conhecida como "template facial".

2.2.1.1 Prova de vida (Liveness Detection)

A prova de vida, ou *liveness detection*, é uma etapa crítica no processo de reconhecimento facial de forma a garantir que a imagem capturada seja de um sujeito vivo e não de uma máscara ou outro tipo de representação estática, como uma fotografia. Isso é essencial para evitar ataques de *spoofing*, nos quais um invasor tenta enganar o sistema de reconhecimento facial com imagens falsas. As técnicas de *liveness detection* podem variar, desde a deteção de movimentos faciais (por exemplo, sorrir) até à análise tridimensional da face usando sensores de profundidade. O objetivo é garantir que a imagem capturada corresponda a um rosto real em tempo real.

2.2.1.2 Deteção do fechar de olhos (Eyes closeness detection)

A deteção do fechar de olhos é uma técnica específica de *liveness detection* que se concentra na verificação se os olhos do sujeito estão fechados de maneira natural, em vez de serem representados por uma imagem estática. Isso é importante porque o movimento dos olhos é uma indicação clara de que a pessoa está viva e presente no momento da captura da imagem.

As técnicas para deteção do fechar de olhos podem envolver a análise de movimentos das pálpebras, a observação de padrões oculares ou o uso de sensores para detetar a presença de olhos abertos e vivos.

2.2.2 Algoritmos de reconhecimento facial

Existem diversos algoritmos utilizados no reconhecimento facial, cada um com suas próprias abordagens e técnicas. Alguns dos principais incluem:

- *Eigenfaces*: Um método clássico que utiliza a análise de componentes principais para extrair características discriminativas do rosto e reduzir a dimensionalidade das imagens faciais;

- Viola-Jones: Um algoritmo de deteção de objetos que utiliza características *Haar* e uma classificação em cascata para identificar rostos em imagens;
- Redes Neurais Convolucionais (CNNs): Modelos de aprendizagem profunda que têm se mostrado muito eficazes em tarefas de reconhecimento facial. As CNNs são capazes de aprender representações hierárquicas complexas das imagens faciais, o que as torna ideais para a aplicação.

Estes algoritmos, combinados com técnicas de pré-processamento e pós-processamento, são essenciais para a construção de sistemas de reconhecimento facial robustos e precisos.

2.3 Biometria de Voz: Fundamentos e Aplicações

A biometria de voz é uma técnica de identificação biométrica que utiliza características únicas da voz de um indivíduo para fins de autenticação e verificação de identidade. Baseia na análise de padrões vocais, como frequência, entoação, duração das sílabas e outros aspetos acústicos que são exclusivos para cada pessoa.

2.3.1 Captura da Voz

A captura da voz envolve a gravação do sinal de áudio da voz de um indivíduo. Esta ação pode ser realizada usando dispositivos como um simples microfone num smartphone, computadores ou em sistemas de gravação dedicados. Durante a captura, são registadas as características únicas da voz do indivíduo, incluindo tom, frequência, padrões de fala e outras características que podem ser utilizadas para identificação biométrica.

2.3.2 Pré-processamento de sinal de voz para biometria

O pré-processamento de sinal de voz é uma etapa crucial na biometria de voz. Envolve um tratamento acústico elaborado, onde são realizadas ações de limpeza e melhoria do sinal de áudio de forma a remover ruídos indesejados e garantir a qualidade dos dados a armazenar. Essa ação pode incluir filtros para eliminar ruídos de fundo, normalização de volume e remoção de artefactos que possam afetar a precisão da análise biométrica.

2.3.3 Extração de Características

Após o pré-processamento, são extraídas características relevantes do sinal de voz que serão utilizadas na identificação ou verificação biométrica. Este processo envolve a análise de parâmetros acústicos, como frequência fundamental, espectro de frequência, tempo de duração das sílabas e padrões fonéticos (ex: entoação). Essas características são então transformadas num formato adequado armazenamento e posterior uso na comparação.

2.3.4 Comparação de voz

Os templates de voz, que são as representações digitais das características da voz de um indivíduo, são comparados para determinar se pertencem à mesma pessoa. Esta comparação é realizada usando algoritmos de comparação que avaliam a similaridade entre os templates de voz previamente recolhidos, com base em métricas específicas. Quanto maior a similaridade entre os templates, maior a probabilidade de pertencerem à mesma pessoa.

2.4 Armazenamento de Template

Os templates resultantes da extração de características são armazenados de forma segura em base de dados. De forma a garantir a segurança desses dados e assim evitar o acesso não autorizado com a finalidade de proteger a privacidade dos indivíduos, serão usadas técnicas de criptografia e proteção de dados, garantindo a integridade e confidencialidade dos templates de voz armazenados.

2.5 Redes Neurais e Aprendizagem Profunda

Estes conceitos fundamentais de redes neurais e aprendizagem profunda são essenciais para se compreender as técnicas modernas de inteligência artificial e suas aplicações, neste caso, ao nosso trabalho.

2.5.1 Redes Neurais (*Neural Networks*)

As redes neurais são modelos computacionais inspirados no funcionamento do cérebro humano. São compostas por camadas de neurônios artificiais em que cada neurônio recebe entradas, realizando cálculos ponderados e produzindo uma saída. As redes neurais podem ter várias camadas, cada uma com neurônios interconectados, permitindo uma representação complexa de dados e aprendizagem de padrões.

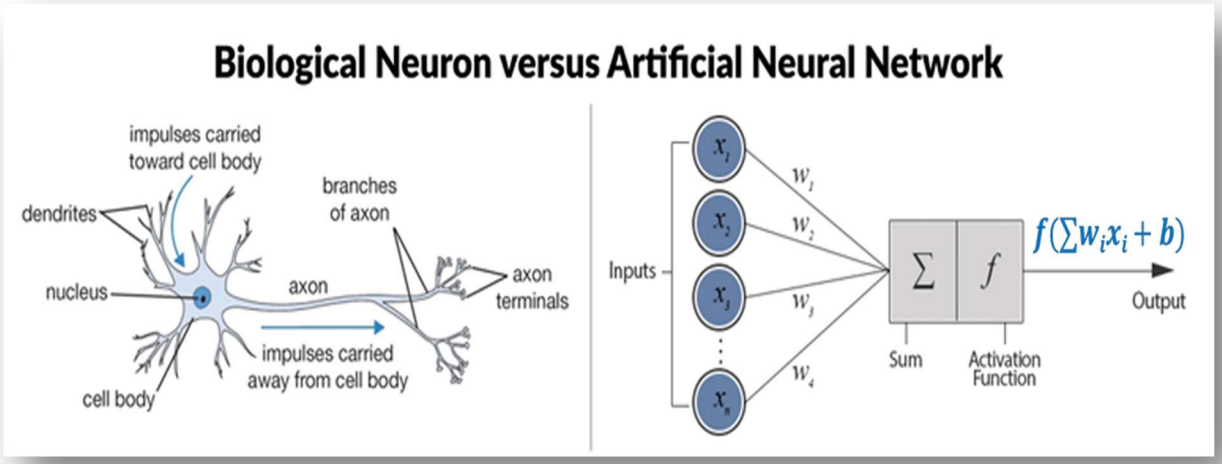

Figura 3 - Analogia entre um neurónio biológico e um neurónio artificial

Fonte: Siddharth Rout [11]

2.5.1.1 Aprendizagem profunda

A aprendizagem profunda, ou *deep learning*, é uma subárea da inteligência artificial que se concentra no treino de redes neurais profundas, de múltiplas camadas. Redes que são capazes de aprender representações hierárquicas de dados complexos, extrair características significativas e realizar tarefas de classificação, previsão ou reconhecimento com alta precisão, figura 4.

A aprendizagem profunda tem sido aplicada com sucesso em uma variedade de domínios, como o processamento de linguagem natural, reconhecimento de voz e facial, assim como muitos outros.

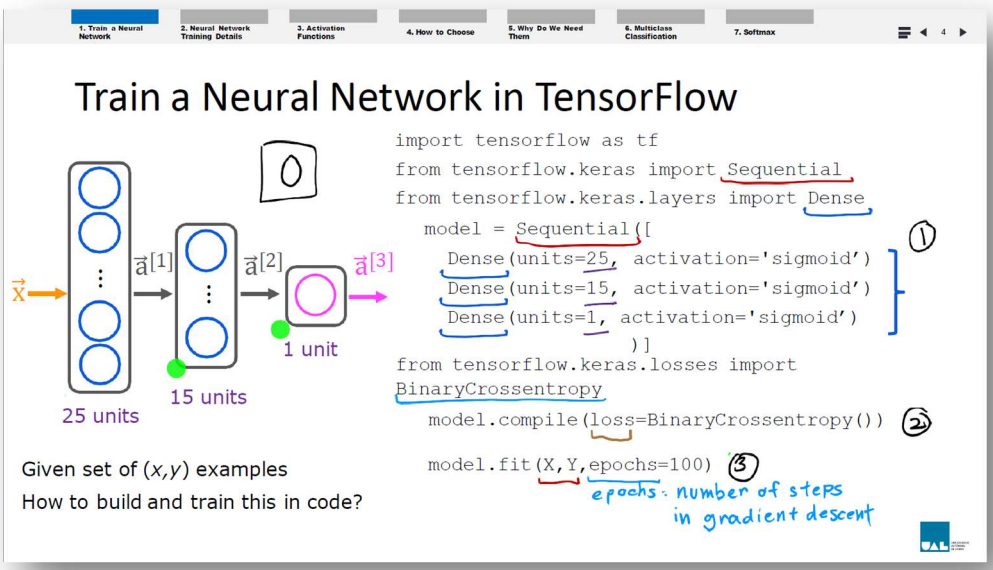

Figura 4 - Redes Neuronais na biblioteca TensorFlow

Fonte: Prof. Dr. Sérgio Ferreira (Inteligência Artificial e Sistemas de Apoio à Decisão)

2.5.2 Rede neural convolucional (CNN)

As redes neurais convolucionais, ou CNNs, são um tipo específico de rede neural projetada para processar dados em forma de grelha, como imagens. São compostas por camadas convolucionais, cujo propósito é aplicar filtros para extrair características locais das imagens, imagens, como bordas, texturas e padrões distintivos. Possuem também camadas de *pooling*, sendo desta forma, a dimensionalidade dos dados reduzida. As CNNs são amplamente utilizadas em tarefas de visão computacional, como reconhecimento de objetos, deteção de facial e segmentação de imagens, devido à sua capacidade de aprender padrões espaciais e hierárquicos, exemplo de calculo na figura 5.

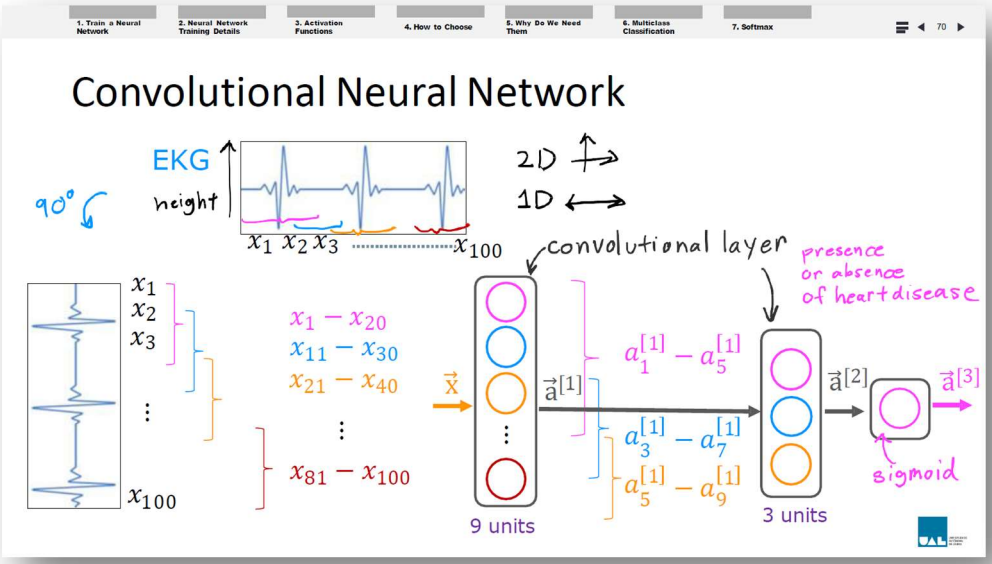

Figura 5 - Redes Neurais Convolucionais (CNNs)

Fonte: Prof. Dr. Sérgio Ferreira (Inteligência Artificial e Sistemas de Apoio à Decisão)

2.5.3 *Embeddings*

Os *embeddings* são representações numéricas de dados de alta dimensão que capturam características semânticas e relações entre os diferentes elementos. Nas redes neurais, os *embeddings* são frequentemente aprendidos durante o treino, onde os dados de entrada são mapeados para um espaço de menor dimensão, onde os pesos das ligações entre os neurônios da rede neural são ajustados de forma iterativa de forma a minimizar a função de perda, geralmente com o objetivo de melhorar o desempenho da rede numa tarefa específica, como classificação ou previsão, mas preservando sempre as relações importantes entre os elementos.

Essas representações densas são úteis em várias tarefas, como processamento de linguagem natural, recomendação de conteúdo e análise de dados, onde é necessário capturar significado semântico e similaridade entre os itens capturados e armazenados.

3 Sistema de Autenticação Digital Anti *Deepfakes*

3.1 Arquitetura Geral do Sistema

Para o nosso sistema de autenticação multifatorial, escolhemos uma arquitetura simples, porém robusta e segura. Conforme ilustrado na Figura 6, o sistema é composto por um servidor de *back-end*, que gerência a lógica de aplicação e os modelos de reconhecimento facial e de voz, e um aplicativo móvel, desenvolvido em *Flutter*, que atua como *front-end*. A comunicação entre o servidor e o aplicativo é feita pela internet, garantindo acessibilidade em dispositivos iOS e Android. A gestão de utilizadores é realizada por um servidor de *backoffice* em *Node.js*, enquanto o *Firebase* é utilizado para armazenar dados dos utilizadores de forma segura e eficiente, oferecendo escalabilidade e flexibilidade.

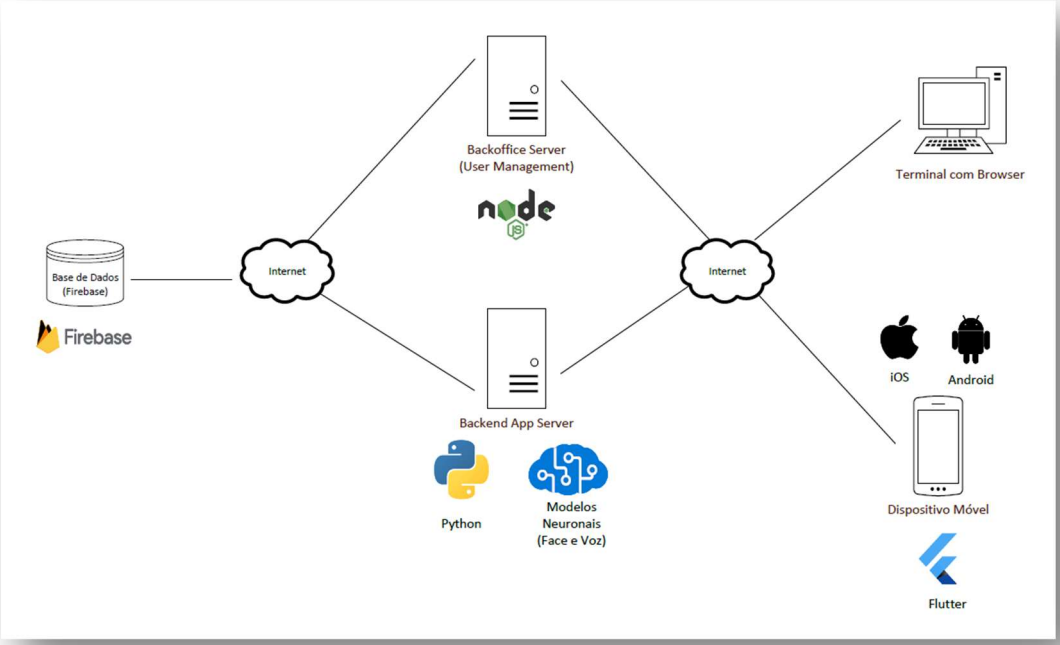

Figura 6 - Arquitetura do Sistema

Fonte: Autores

3.2 Módulo de Reconhecimento Facial

O módulo de reconhecimento facial é uma parte essencial do nosso sistema de autenticação anti *deepfake*, sendo responsável pela análise e identificação de características faciais únicas de cada utilizador. Este processo é crucial para garantir a autenticidade e a segurança durante a autenticação.

3.2.1 Deteção e localização de faces

Esta técnica avançada, conforme ilustrado na Figura 7, deteta se um utilizador está presente sem exigir ações ou gestos explícitos, analisando um vídeo ou imagem única para verificar características como luz, textura da pele e micro movimentos.

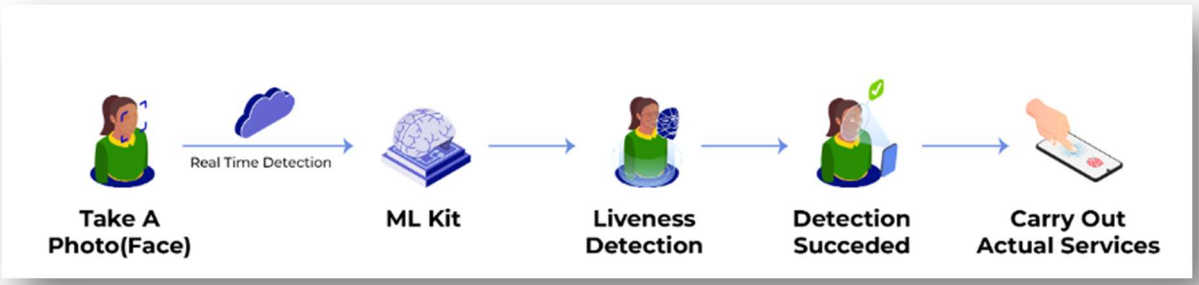

Figura 7 - Fluxo de prova de vida

Fonte: https://decentro.tech/blog/liveness-check/

O funcionamento da prova de vida pode ser simplificado assim:

- Reconhecimento facial: Captura e analisa características faciais para criar um modelo biométrico.
- Perceção de profundidade: Utiliza mapeamento de profundidade 3D para avaliar informações espaciais do rosto.
- Análise de movimento: Solicita ao usuário ações como piscar ou sorrir, analisando a naturalidade dos movimentos.
- Análise de textura e cor: Examina padrões de textura e cor do rosto para garantir realismo.
- Desafios aleatórios: Apresenta desafios imprevisíveis para evitar falsificações.
- Biometria comportamental: Analisa padrões únicos de movimentos do usuário.

- Aprendizado de máquina e IA: Usa algoritmos para melhorar continuamente a precisão da verificação.
- Análise em tempo real: Fornece feedback imediato durante a verificação, garantindo participação ativa do utilizador.

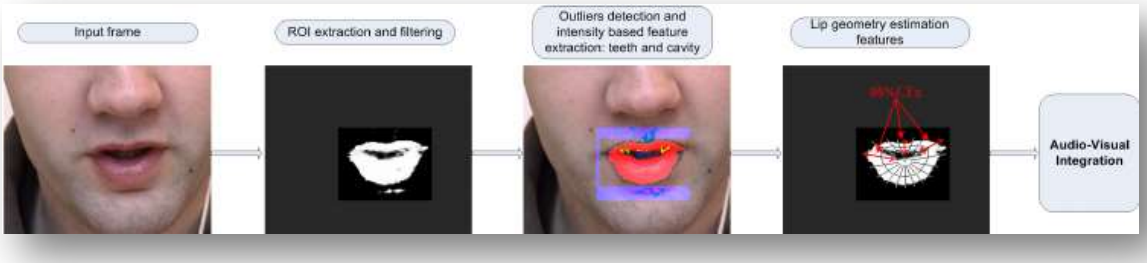

Figura 8 - Extração de prova de vida [12]

Fonte: https://www.researchgate.net/figure/Signal-processing-pipeline-for-the-audio-visual-speech-recognition-system-based-on-LGE_fig2_225820483 [12]

3.2.2 Extração de características faciais

A extração de características faciais envolve várias etapas, desde a captura da imagem até a criação de uma representação algorítmica do rosto. Este processo permite que o sistema diferencie indivíduos e detete tentativas de falsificação.

Etapas de extração das características faciais:

- Pré-processamento de imagem:
  - Captura da imagem: Utilizando a câmara frontal do dispositivo móvel.
  - Conversão para escala de cinza: A imagem é convertida para escala de cinza, para aumentar o desempenho computacional e simplificar o processamento.
  - Normalização: A imagem é normalizada para ajustar a iluminação, garantindo consistência nas características extraídas, independentemente das condições de iluminação.
- Deteção de rosto:
  - *'OpenCV'*: Com a utilização do *OpenCV* é possível detetar a presença do rosto na imagem.

    - Alinhamento do rosto: Após a deteção, o rosto é alinhado para corrigir inclinações e garantir que os olhos, nariz e boca estejam nas posições esperadas. Isso é feito utilizando pontos de referência faciais.
- Extração de pontos faciais
    - *'Dlib':* A biblioteca *dlib* é utilizada para identificar pontos de referência faciais, como os cantos dos olhos, extremidades dos lábios e a ponta do nariz.
    - Precisão dos pontos: Os pontos faciais são ajustados para garantir alta precisão, utilizando técnicas de refinamento disponíveis na *dlib*.
- Prova de vida
    - Movimentos faciais: O sistema verifica movimentos faciais como piscar de olhos, abrir e fechar a boca para garantir que o rosto capturado é de uma pessoa real e não uma imagem estática.
    - Análise temporal: Captura de múltiplas imagens ao longo do tempo para confirmar que os movimentos são naturais e consistentes com um rosto real.

As tecnologias utilizadas aqui são:

- *OpenCV*
- *Dlib*
- *DeepFace*
- *Tensorflow*

A utilização das bibliotecas acima mencionadas, permite uma deteção precisa e eficiente de rostos, assegurando a autenticidade dos utilizadores. A metodologia empregada, que inclui prova de vida através de movimentos faciais, aumenta significativamente a robustez contra tentativas de falsificação, garantindo a segurança e a confiabilidade do sistema.

3.2.3 Reconhecimento facial com redes neurais convolucionais

O reconhecimento facial com redes neurais convolucionais (CNNs) é uma técnica avançada que utiliza modelos de aprendizagem profunda para identificar e verificar a identidade de indivíduos com alta precisão. As CNNs são especialmente eficazes em tarefas de visão

computacional devido à sua capacidade de extrair características complexas e invariantes de imagens.

As camadas de *pooling* reduzem a dimensionalidade dos mapas de características, mantendo as informações mais importantes, onde máximo *pooling* é uma técnica comum que seleciona o valor máximo de uma região, reduzindo a resolução espacial.

As CNNs são compostas por várias camadas que transformam a imagem de entrada em uma representação hierárquica. As camadas convolucionais aplicam filtros à imagem de entrada para detetar características locais, como bordas, texturas e padrões, com cada filtro treinado para responder a diferentes características visuais.

As camadas de normalização normalizam os valores de ativação, ajudando na estabilização do processo de treinamento e na convergência mais rápida da rede. Por fim, as camadas completamente conectadas, que são as finais da CNN, conectam cada neurônio a todos os neurônios da camada anterior, permitindo a combinação de características extraídas para realizar a classificação ou a identificação.

As camadas completamente conectadas, que são as finais da CNN, conectam cada neurônio a todos os neurônios da camada anterior, permitindo a combinação de características extraídas para realizar a classificação ou a identificação.

Para o reconhecimento facial, as CNNs são treinadas com um grande conjunto de dados de rostos humanos, permitindo que aprendam a identificar padrões faciais únicos e discriminativos.

As imagens faciais são normalizadas e alinhadas para garantir consistência na posição e no tamanho dos rostos. Logo de seguida as características faciais discriminativas são extraídas de várias camadas convolucionais, capturando detalhes essenciais para a identificação individual, essas características são transformadas em vetores de alta dimensão chamados *embeddings* faciais, que representam a identidade do rosto de maneira única.

As CNNs têm se mostrado altamente precisas na identificação de rostos, superando métodos tradicionais devido à sua capacidade de aprender características complexas. Além disso, as redes convolucionais são robustas contra variações de iluminação, ângulo e expressão facial, tornando-as ideais para aplicações em ambientes variados. Também são escaláveis, pois modelos de CNN podem ser treinados com grandes volumes de dados, melhorando continuamente com mais exemplos e aumentando a precisão conforme novos dados são adicionados.

3.3 Módulo de Biometria de Voz

O módulo de biometria de voz é uma componente essencial do nosso sistema de autenticação anti *deepfake*. Utilizando a biblioteca *librosa* para a extração de características acústicas e a tecnologia de reconhecimento de voz da Azure, garantimos um processo de verificação preciso e seguro.

Para a autenticação, o sistema captura a voz do utilizador enquanto este repete uma frase exibida no ecrã. As características acústicas, como os coeficientes MFCC, o *chromagram*, o mel *spectogram* e o *spectral contrast*, são extraídas utilizando a biblioteca *librosa*. Esses dados são comparados com os registos existentes na base de dados para verificar a identidade do utilizador.

Aplicamos a tecnologia Azure para converter a fala em texto, confirmando se a frase dita corresponde à esperada.

3.3.1 Pré-processamento do sinal de voz

O pré-processamento do sinal de voz é uma etapa no módulo de biometria de voz, responsável por preparar os dados de áudio para a análise e extração de características que possibilitam a autenticação segura e precisa dos usuários. Este processo, tal como demonstra a figura 9, envolve várias etapas que transformam o sinal de voz bruto em um formato adequado para a aplicação de técnicas avançadas de reconhecimento.

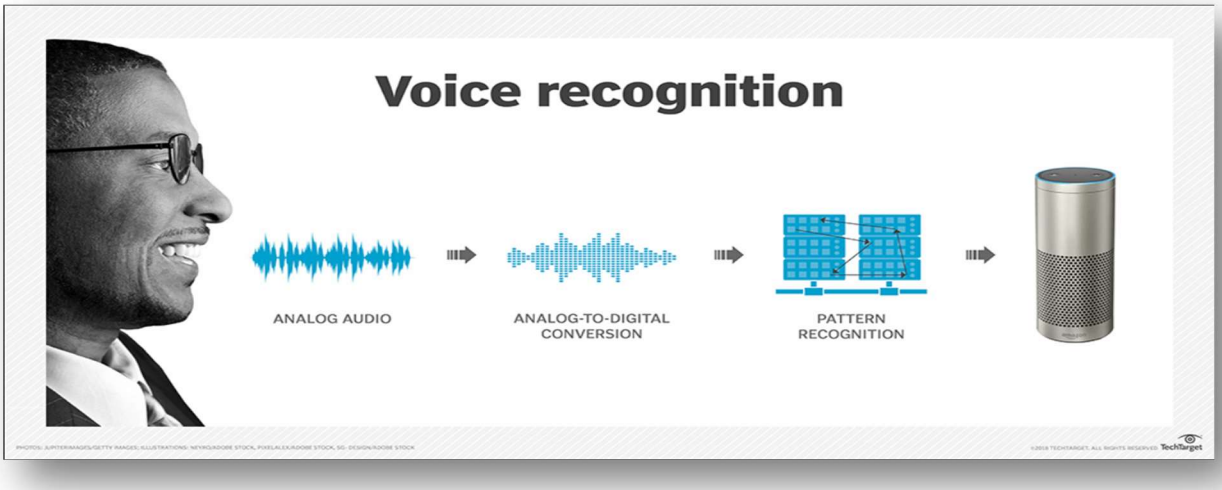

Figura 9 - Fluxo de reconhecimento de voz

Fonte: https://www.techtarget.com/searchcustomerexperience/definition/voice-recognition-speaker-recognition

O sinal de voz é recolhido utilizando dispositivos de entrada (microfones), a qualidade do áudio é fundamental, pois ruídos e distorções podem afetar negativamente o desempenho do sistema. Para isto, são aplicados filtros para redução de ruído e normalização do volume, garantindo que o sinal de entrada esteja em um nível consistente e livre de interferências indesejadas. Após esta filtragem o sinal de voz continuo é dividido em segmentos menores (*frames*). Esta divisão permite a análise do sinal de voz em pequenas porções temporais, facilitando a captura de características dinâmicas da fala que mudam rapidamente.

3.3.2 Extração de características vocais

Após a segmentação, são extraídas as características essenciais do sinal de voz.

Esta etapa, envolve a transformação do sinal de voz pré-processado em representações que podem ser usadas para a análise e comparação.

Para isso, utilizamos a biblioteca *librosa* que disponibiliza inúmeras tecnologias que nos permitem executar esta análise de características da melhor forma.

- MFCC (*(Mel-frequency cepstral coefficients*)
    - Os MFCCs capturam as propriedades espectrais da voz humana de forma que se alinhem com a perceção auditiva humana. Eles são obtidos através da transformação do espectrograma de potência do sinal de voz em uma escala Mel. Esta transformação resulta em coeficientes que representam a forma do espectro de potência, permitindo a captura de características relevantes para a identificação de vozes.
- *Mel Spectogram*
    - O *Mel Spectogram* é outra característica vital que representa a distribuição de energia do sinal de voz ao longo do tempo e das frequências. Utilizando a escala Mel, que imita a perceção humana das frequências sonoras, o *Mel Spectogram* facilita a visualização e análise de padrões de frequência que são exclusivos para cada indivíduo. Esta

representação é especialmente útil para capturar informações detalhadas sobre o timbre e a tonalidade da voz.

- *Spectral Contrast*
    - O *Spectral Contrast* mede a diferença de amplitude entre picos e vales em diferentes bandas de frequência do espectro de áudio. Esta característica é importante para distinguir entre regiões de alta e baixa energia no sinal de voz, ajudando a diferenciar sons harmoniosos de ruídos e a identificar características específicas do timbre vocal de um indivíduo.
- *Chromagram*
    - Esta característica, representa a energia das frequências correspondentes às 12 notas da escala cromática, independentemente da oitava. Esta característica é útil para analisar a tonalidade e a harmonia do sinal de voz, fornecendo informações adicionais sobre o conteúdo espectral da voz. O *chromagram* é particularmente útil para capturar nuances musicais e timbres da voz humana.
- *Dynamic time warping* (DTW)
    - Para comparar as características extraídas de diferentes amostras de voz, utilizamos a técnica de DTW. O DTW é um algoritmo que mede a similaridade entre duas sequências temporais que podem variar em velocidade e ritmo. Ele alinha as sequências de forma não linear, permitindo uma comparação precisa mesmo quando as amostras de voz possuem diferenças temporais. Esta técnica é essencial para comparar as características vocais de uma amostra de teste com uma amostra de referência.
- Azure *Speech* SDK
    - Além das técnicas mencionadas, utilizamos o Azure *Cognitive Services* para reconhecimento de fala, especificamente o Azure *Speech* SDK. Este serviço converte a fala em texto, permitindo a verificação da frase pronunciada pelo utilizador. A integração com o Azure *Speech* SDK garante que a transcrição da fala seja precisa e confiável, complementando a análise de características vocais

3.3.3 Classificação de voz com redes neurais convolucionais.

A classificação de voz com redes neurais convolucionais (CNNs) é uma abordagem inovadora e eficaz para o reconhecimento e autenticação de voz.

As CNNs, amplamente utilizadas em tarefas de visão computacional, têm se mostrado igualmente eficazes na análise de sinais de áudio devido à sua capacidade de extrair e aprender características complexas e hierárquicas dos dados.

No contexto da classificação de voz, o sinal de áudio é primeiramente transformado em uma representação espectral, como um *Mel Spectogram*, que pode ser interpretado como uma imagem. Este espectrograma serve como entrada para a CNN, que processa os dados de áudio de maneira similar ao processamento de imagens.

As camadas convolucionais da CNN extraem características locais do espectrograma, capturando informações sobre frequências, harmônicos e padrões temporais na voz. Estas características são então combinadas e refinadas ao longo das camadas da rede, resultando em uma representação que pode ser usada para a classificação.

O treino da CNN envolve a alimentação da rede com um grande conjunto de dados de voz com rotulo, onde cada exemplo de treino está associado a um rótulo indicando a identidade do falante ou outra classe de interesse. Os pesos dos filtros convolucionais são ajustados para minimizar a diferença entre as previsões da rede e os rótulos verdadeiros. Este processo é guiado por algoritmos de otimização, como o gradiente descendente.

Após o treino, a CNN pode ser usada para classificar novas amostras de voz. Quando uma nova amostra de voz é fornecida à rede, a CNN processa o espectrograma correspondente e produz uma previsão indicando a identidade do falante ou se a amostra de voz corresponde a um utilizador autenticado.

As CNNs oferecem uma capacidade robusta de extração e aprendizagem de características vocais, resultando em um sistema de autenticação preciso e eficiente. Integradas com outras técnicas e ferramentas de reconhecimento facial e biometria de voz, as CNNs fortalecem significativamente a segurança e a confiabilidade do sistema, protegendo contra tentativas de falsificação e ataques de *deepfake*.

3.4 Módulo de Fusão do Reconhecimento facial e Biometria de Voz

3.4.1 Técnicas de fusão de autenticação

O diagrama apresentado na figura 10 [16], ilustra um processo de análise de áudio para autenticação. O áudio é transformado numa espectrograma e LFCC (Coeficientes de Filtro de Frequência), que alimentam uma rede SE-DenseNet para extração de características de fala.

Simultaneamente, um codificador de voz extrai características faciais. Esses vetores são concatenados e passam por uma camada de atenção, que foca em aspetos importantes para a classificação. Finalmente, uma rede de classificação determina se a amostra é genuína ou uma tentativa de falsificação.

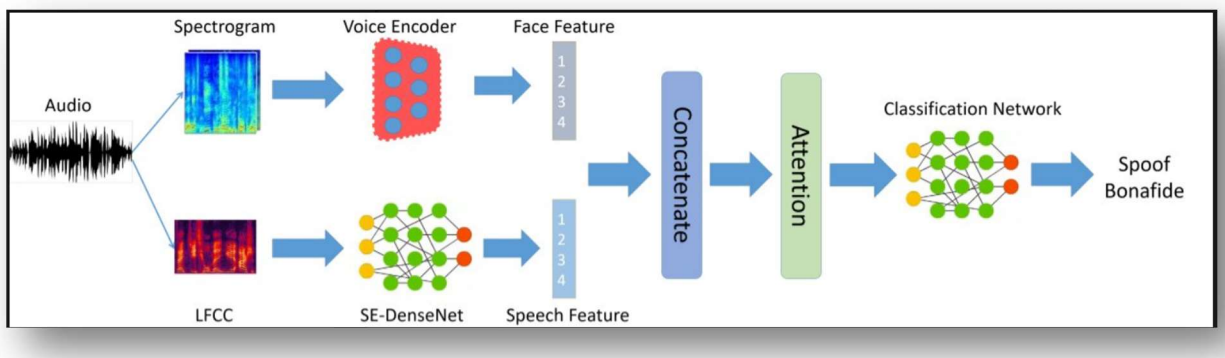

Figura 10 - Estrutura prova de vida e voz [16]

Fonte: https://techxplore.com/news/2023-05-physiological-physical-feature-fusion-automatic-voice.html

3.4.2 Estratégia de Autenticação Integrada (facial e voz)

O diagrama de sequência, conforme figuras 11 e 12, detalha o fluxo de autenticação multifatorial, destacando a interação entre o utilizador, o aplicativo móvel e os componentes de *back-end*. Inicialmente, o utilizador realiza o registo, enviando um vídeo que é processado pelo *back-end* para reconhecimento facial utilizando o *Facenet 512*. Simultaneamente, amostras de áudio são enviadas para reconhecimento de voz. Após a validação das expressões faciais e da voz, é realizada uma prova de vida para garantir a autenticidade. Os vetores de reconhecimento facial e de voz são registados na base de dados do *Firebase*, assegurando uma autenticação segura e eficiente.

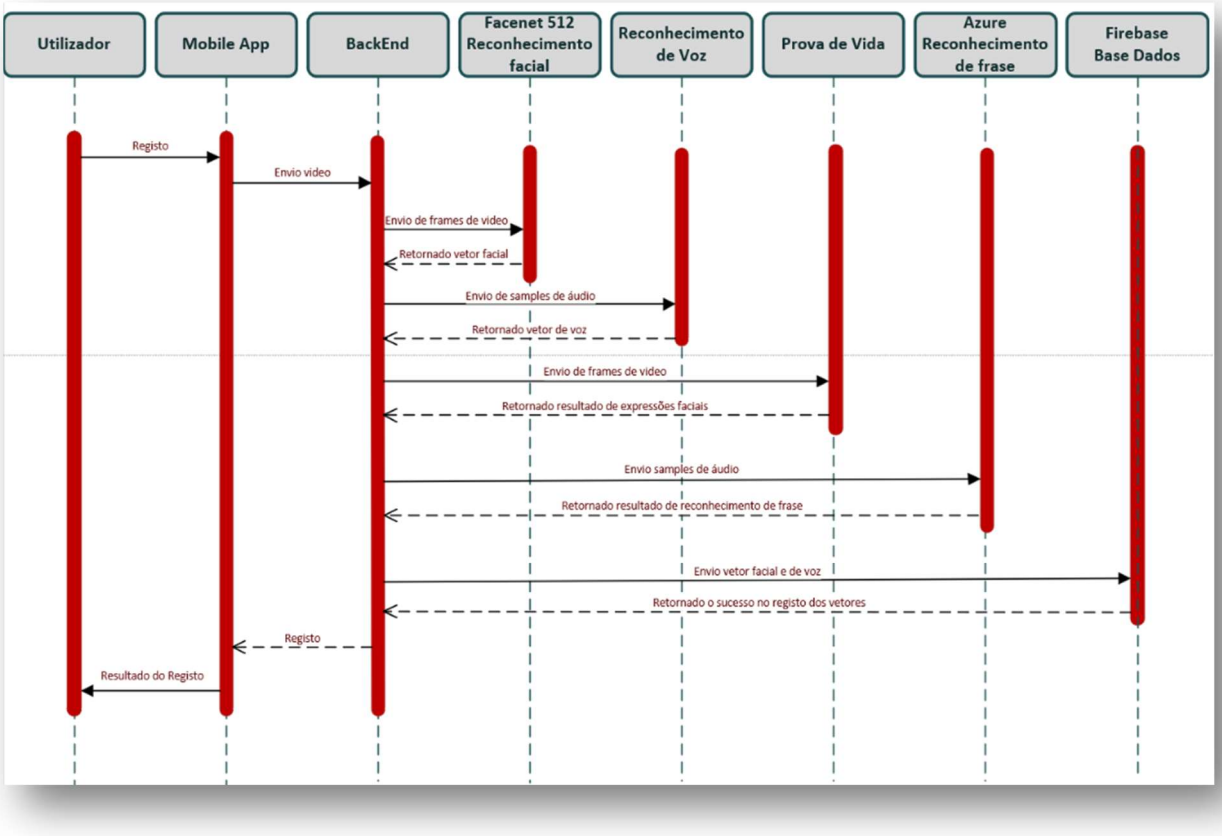

Figura 11 - Diagrama de sequência - Registo de utilizador

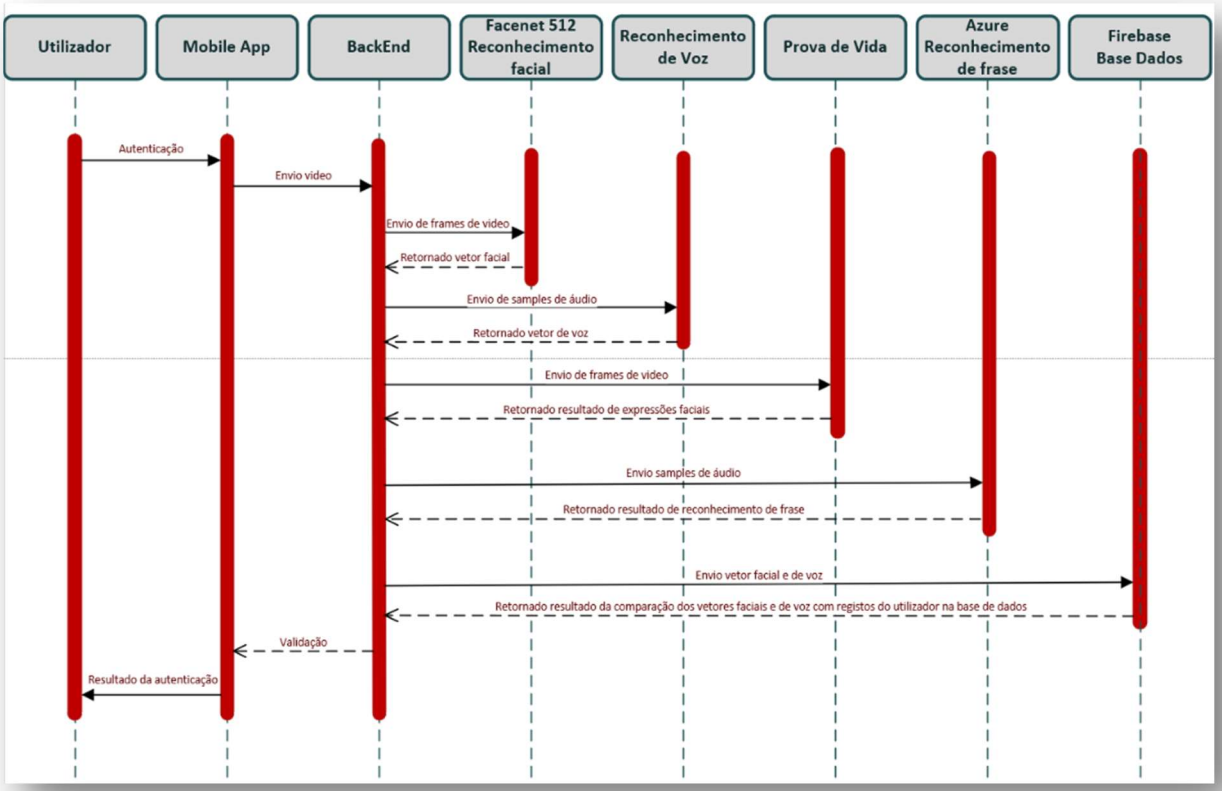

Figura 12 - Diagrama de sequência - Autenticação

Fonte: Autores

4 Implementação e Experimentação

4.1 Base de Dados e Ferramentas

Para que a implementação desta aplicação possa ser feita com sucesso, tivemos de utilizar uma série de aplicações e conexões de forma a conseguirmos a integração dos diversos componentes utilizados para implementação do mesmo.

Tal como explicado previamente a nossa aplicação tem como objetivo a autenticação de utilizadores através de vários fatores de autenticação tais como, reconhecimento facial e biometria de voz. Para que seja possível extrair, processar e carregar as informações acerca do utilizador em questão, temos então de utilizar uma série de ferramentas que serão descritas abaixo:

4.1.1 Firebase

O *Firebase*, é uma plataforma de desenvolvimento de aplicações móveis e web disponibilizada pela lista de aplicações da google, que permite o armazenamento de dados biométricos de forma segura em nuvem (*cloud*), em tempo real. O *Firebase* garante integridade e confidencialidade dos dados que estão inseridos na base de dados.

4.1.2 Flutter

O *Flutter* é uma estrutura de desenvolvimento de interface do utilizador, disponibilizada na lista de aplicações da google. Esta estrutura permite criar e desenvolver aplicações móveis para Android e iOS a partir de uma base de código exclusiva. Uma das características desta estrutura é a funcionalidade de recarregamento a quente, permite assim à equipa de desenvolvimento, fazerem a verificação das alterações feitas no código em tempo real.

4.1.3 Google ML Kit

O Google ML Kit é uma plataforma disponibilizada pela Google que fornece ferramentas e APIs para incorporar funcionalidades de aprendizagem de máquina (Machine Learning) em aplicações movéis. Esta plataforma é utilizada para equipas de desenvolvimento

que desejam incorporar aprendizagem de máquina nas suas aplicações, mas que não tenham a necessidade de um conhecimento profundo no mesmo.

O Google ML Kit apresenta alguns componentes que devem ser considerados:

1. Vision APIs – Possibilitam a incorporação de recursos de visão computacional nas aplicações. As suas principais funcionalidades são:
   - Deteção de rosto – Identifica rostos em imagens e vídeos, fornecendo informação sobre a localização dos olhos, nariz, boca e contornos do rosto.
   - Etiquetas em imagens – Classifica as imagens em categorias especificas, estas etiquetas baseiam-se no conteúdo da própria imagem
   - Segmentação de imagem – Distingue e segmenta as diferentes partes de uma imagem, como pessoas, objetos e o fundo da própria imagem.

2. Linguagem Natural de APIs
   - Reconhecimento de entidades – Identifica e classifica as entidades mencionadas em um texto, como nomes de pessoas, locais, datas, etc…
   - Análise de sentimento – Avalia o sentimento por trás de um texto, identificando se é positivo, negativo ou neutro.

3. Modelos customizados: O Google ML Kit disponibiliza vários modelos de aprendizagem de máquina personalidades às necessidades do programador. Estes modelos podem ser treinados utilizando o Tensorflow Lite e depois podem ser integrados à aplicação desenvolvida. Esta funcionalidade de modelos personalizados inclui:
   - Carregamento de modelos – Facilita a inclusão de modelos treinados personalizados nas aplicações.
   - Ajuste de desempenho – Otimiza os modelos para melhor desempenho em dispositivos móveis.
   - Testes – Permite testar diferentes versões de modelos

Podemos ver várias vantagens na utilização desta estrutura, tais como, a facilidades de uso onde as APIs são fáceis de integrar; as possibilidade de utilização em mais do que uma plataforma, (Android e iOS); o desempenho local, onde muitos dos modelos funcionam diretamente no dispositivo garantindo assim respostas rápidas e melhor proteção da privacidade do utilizador; atualizações constantes, visto que é uma estrutura Google, então a necessidade de melhoria e adição de novas funcionalidades no ML Kit torna-se constante de forma a suportar as equipas de desenvolvimento.

4.1.4 Facenet 512

O *FaceNet* é um sistema de reconhecimento facial desenvolvido pela Google que utiliza *deep-learning* para gerar vetores de *embeddings* faciais. Esses *embeddings* são representações matemáticas de rostos, que podem ser usados para diversas tarefas, como reconhecimento e verificação facial. A variante *FaceNet 512* refere-se ao uso de vetores de 512 dimensões para representar essas *embeddings*, oferecendo um equilíbrio entre precisão e eficiência computacional.

Este sistema é altamente eficaz em identificar pessoas com base nas suas características faciais. Ele pode ser utilizado em sistemas de segurança, controlo de acessos, redes sociais e muito mais. O processo de reconhecimento envolve comparar o vetor de *embeddings* de um rosto desconhecido com uma base de dados de vetores conhecidos. Além de reconhecer rostos, o *FaceNet 512* pode ainda verificar a identidade de uma pessoa comparando dois rostos e determinando se eles pertencem à mesma pessoa. Isto é especialmente útil em sistemas de autenticação biométrica, onde é necessário confirmar a identidade do utilizador. Calcula a similaridade entre dois rostos, gerando uma pontuação que indica o quão parecidos estes são.

Este modelo pode agrupar rostos semelhantes sem qualquer identificação prévia. Este processo de agrupamento é feito com base na proximidade dos vetores de *embeddings*, permitindo a organização de grandes conjuntos de imagens faciais em clusters de indivíduos.

O *FaceNet* como sistema de reconhecimento facial, tem algumas características técnicas que serão mencionadas abaixo:

- *Embeddings* de 512 Dimensões
    i. Os vetores de 512 dimensões utilizados pelo FaceNet 512 são um compromisso entre a precisão do reconhecimento e a eficiência computacional. Esses *embeddings* capturam de forma robusta as características únicas de cada rosto, permitindo uma alta taxa de acerto nas tarefas de reconhecimento e verificação.
- Como é treinado o *FaceNet 512*?
    i. O *FaceNet* é treinado usando uma técnica chamada *"triplet loss*", que ajuda a otimizar as distâncias entre os vetores de *embeddings*. Isso envolve a seleção de três imagens durante o processo de treino: uma âncora, uma imagem positiva (do mesmo indivíduo) e uma imagem negativa (de um indivíduo diferente). O objetivo é minimizar a distância entre a âncora e a positiva e maximizar a distância entre a âncora e a negativa.

- Resiliência a variações
    i. Este sistema é projetado para ser resiliente a várias condições de imagem, como iluminação, ângulo e expressões faciais diferentes. É possível através do treinamento com um conjunto de dados diversificado e robusto.
- Desempenho e eficiência
    i. A arquitetura do *FaceNet 512* é otimizada para oferecer um desempenho elevado em termos de precisão e velocidade. Embora utilize *embeddings* de alta dimensão, ele é eficiente o suficiente para ser implementado em tempo real em diversos dispositivos e aplicações.

Em suma, o *FaceNet 512* é uma ferramenta de reconhecimento facial que combina precisão, eficiência e resiliência. As suas funcionalidades robustas e características técnicas avançadas tornam um sistema ideal para uma ampla gama de aplicações, desde segurança e autenticação até redes sociais e e-commerce.

4.1.5 TensorFlow

TensorFlow é uma plataforma de código aberto para aprendizagem de máquina desenvolvida pelo *Google*. Lançada inicialmente em 2015, a plataforma rapidamente tornou-se uma das ferramentas mais populares para desenvolver e treinar modelos de aprendizagem de máquina e *deep learning*. Esta infraestutura tem como objetivo facilitar a construção e o treino de modelos de aprendizagem de máquina e *deep learning* de maneira eficiente e escalável. Permite a criação de modelos desde o nível básico até aplicações complexas de IA, proporcionando uma API flexível e uma infraestrutura otimizada para execução em diversos ambientes e é também baseado em uma arquitetura de gráficos de fluxo de dados, onde operações matemáticas são representadas, figura 13.

Figura 13 – Logotipo TensorFlow

Fonte: https://en.wikipedia.org/wiki/TensorFlow

4.1.6 TensorFlow Lite

TensorFlow Lite é uma versão otimizada do *TensorFlow*, projetada especificamente para permitir a execução eficiente de modelos de aprendizagem de máquina em dispositivos móveis. Desenvolvido pelo *Google*, o *TensorFlow Lite* é uma parte crucial do ecossistema *TensorFlow*, voltada para trazer a potência da aprendizagem de máquina para dispositivos com recursos limitados, como *smartphones*, *tablets*, dispositivos IoT e outros sistemas.

Esta infraestrutura tem como objetivo proporcionar uma solução eficiente de bom desempenho para a indução de modelos de aprendizagem de máquina em dispositivos com a capacidade de processamento e memória limitadas. Projetado para maximizar a velocidade de inferência e minimizar o uso de recursos, tornando-o ideal para aplicações em tempo real.

Esta plataforma é composta por dois componentes principais:

- *TensorFlow Lite Conversor*: Esta ferramenta converte modelos treinados em *TensorFlow* (ou outras bibliotecas compatíveis) em um formato otimizado para execução em dispositivos móveis e incorporados (IoT), focado em eficiência e menor uso de recursos. O processo de conversão inclui várias técnicas de otimização, que reduzem o tamanho do modelo e aumentam a eficiência computacional.

- TensorFlow Lite Interpretador: Este é o tempo de execução dos modelos convertidos no dispositivo. É leve e eficiente, projetado para utilizar o mínimo de recursos computacionais possíveis e de memória. O interpretador suporta uma variedade de operações de aprendizagem de máquina, e pode ser estendido com operações customizadas conforme necessário, esquematizado na figura 14.

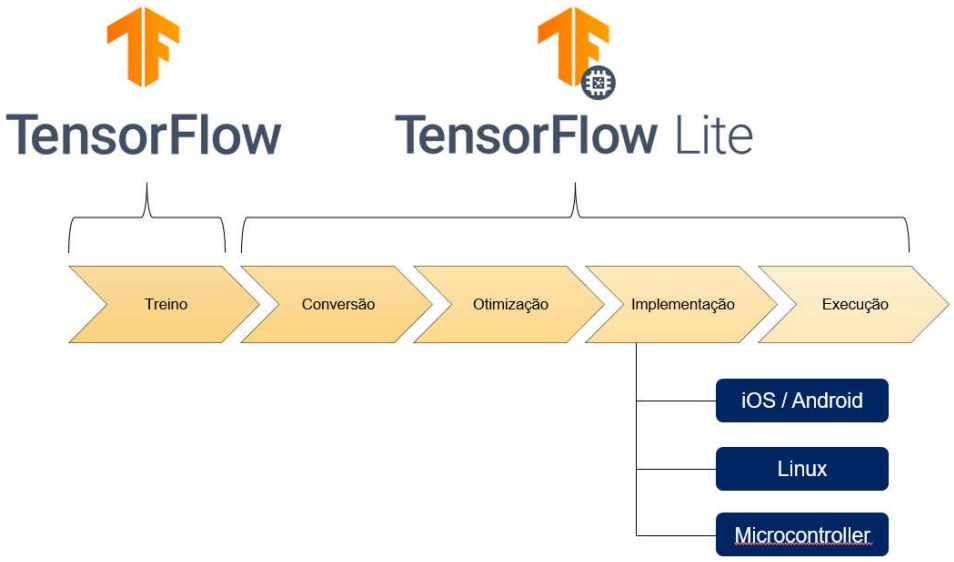

Figura 14 - TensorFlow Lite

Fonte: Autores

4.1.7 Python

Python é uma linguagem de programação de alto nível, interpretada e de propósito geral, conhecida por sua sintaxe clara e legibilidade. Esta linguagem foi projetada para enfatizar a legibilidade do código e a simplicidade, permitindo que as equipas de desenvolvimento escrevam código claro e conciso. Além disso, *Python* foi criado para ser altamente extensível, permitindo a integração com outras linguagens e bibliotecas, o que a torna uma escolha ideal para uma ampla variedade de aplicações.

A utilização do *Python* neste projeto foi sem dúvida crucial. É aqui a base de processamento, transformação e carregamento de toda a informação necessária para detetar e mitigar vídeos falsos gerados por técnicas de *deepfake.*

4.2 Descrição geral

O crescente uso de tecnologias de *deepfake*, que permitem a criação de vídeos e áudios falsificados de maneira altamente realista, tem gerado preocupações significativas em relação à segurança e autenticidade digital. Em resposta a essa ameaça emergente, desenvolvemos um sistema de autenticação robusto que utiliza reconhecimento facial e biometria de voz para verificar a identidade dos utilizadores e prevenir fraudes. Este capítulo descreve a visão geral do sistema, incluindo os componentes principais, as tecnologias utilizadas e o fluxo de trabalho.

Nosso sistema é composto por vários componentes integrados que trabalham juntos para realizar a autenticação de forma segura e precisa.

A interface principal para os utilizadores é uma aplicação movel desenvolvida em *Flutter*. Esta aplicação fornece uma plataforma que permite que o sistema funcione em dispositivos iOS e Android. A aplicação recolhe dados biométricos do utilizador (imagens faciais e áudio de voz) e os envia para o *back-end* para verificação.

Utilizamos *Firebase* para o armazenamento da informação. Aqui será onde os dados biométricos da face e da voz irão ficar armazenados para mais tarde serem utilizados como meio de comparação e validação.

A analise, processamento e carregamento dos dados biométricos é realizado utilizando algoritmos e bibliotecas de aprendizagem de máquina e aprendizagem profunda. Escolhemos o *python* como a linguagem principal utilizada para a implementação e desenvolvimento desses algoritmos.

O fluxo de trabalho do sistema de autenticação anti *deepfake* pode ser resumido nas seguintes etapas:

1. Recolha de dados: O utilizador abre a aplicação móvel e é solicitado a capturar um video facial e gravar um breve áudio.
2. Envio de dados: Os dados biométricos capturados são enviados para o *back-end* via uma conexão segura.

3. Processamento de dados: No *back-end*, os dados são pré-processados e analisados. As características faciais e vocais são extraídas e comparadas com os modelos de referência armazenados.
4. Verificação e deteção: Utilizando modelos treinados, o sistema verifica a autenticidade dos dados biométricos. Se os dados forem validados e não forem detetadas manipulações de *deepfake*, a autenticação é aprovada.
5. Resposta ao utilizador: O resultado da autenticação é enviado de volta para a aplicação móvel, informando o utilizador sobre o sucesso ou falha da autenticação.

4.2.1 Requisitos específicos

4.2.1.1 Requisitos de interface interna/ externa

API é a sigla inglesa para *Application Programming Interface*, ou interface de programação de aplicações, são conjuntos de ferramentas, definições e protocolos que permitem a criação de aplicações e comunicações entre diferentes softwares, ver diagrama da figura 15. São amplamente usadas em aplicações *web*, aplicativos móveis, serviços em nuvem e muitas outras soluções de software de forma a permitir a comunicação entre diferentes sistemas, conectando soluções e serviços, sem a necessidade de saber como os elementos foram implementados. Elas funcionam como se fossem contratos, representando um acordo entre as partes interessadas. Se uma dessas partes enviar uma solicitação remota estruturada de uma forma específica, isso determinará como a aplicação da outra parte responderá. [6]

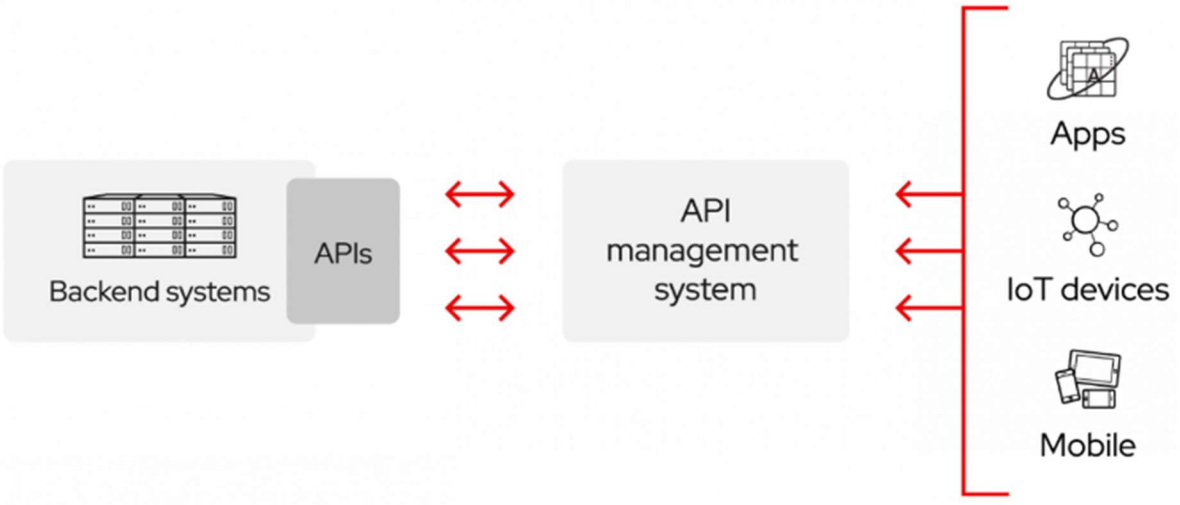

Figura 15 - Diagrama de uma API

Fonte: https://www.redhat.com

4.2.1.2 Requisitos funcionais

A aplicação U@LTech foi desenvolvida como parte de um projeto universitário, visando proporcionar uma ferramenta eficiente e inovadora para os utilizadores. Embora a aplicação tenha sido concebida para funcionar da melhor forma possível, há certos requisitos funcionais que precisam ser atendidos para assegurar o seu desempenho ideal. Este documento detalha esses requisitos e fornece orientações para os utilizadores.

1. Atualização de software
    a. Para garantir a compatibilidade e o desempenho da aplicação U@LTech, é essencial que os dispositivos dos utilizadores estejam equipados com a versão mais recente do sistema operacional. Recomendamos que os utilizadores mantenham seus dispositivos atualizados regularmente.
2. Instalação da aplicação Test Flight
    a. Antes de utilizar a U@LTech, os utilizadores devem instalar o Test Flight, uma plataforma de teste de aplicações. Este sistema permite que a aplicação seja distribuída aos utilizadores convidados sem nenhum custo associado. Instruções detalhadas para a instalação do Test Flight serão fornecidas no convite enviado aos utilizadores.
3. Conexão à internet
    a. A aplicação requer uma conexão estável à internet para funcionar corretamente. Isso é fundamental para o upload e download de dados, autenticação de utilizadores e comunicação com os servidores. Infelizmente a nossa aplicação não funciona com a network da universidade autónoma de Lisboa pois o servidor encontra-se alocado nas portas 5555 ou 5556 (bloqueadas pela *firewall* da universidade).
4. Hardware necessário
    a. Câmara frontal: O dispositivo deve possuir uma câmara frontal funcional, que será utilizada para gravação de vídeos como parte do processo de verificação e análise.
    b. Microfone: O dispositivo deve possuir um microfone operacional para capturar o áudio enquanto o utilizador fala a frase de verificação.
5. Permissões da aplicação

a. Acesso à câmara: A aplicação solicitará permissão para aceder à câmara frontal. É imprescindível conceder esta permissão para que a gravação de vídeo seja possível.
b. Acesso ao microfone: A aplicação solicitará permissão para aceder ao microfone. É necessário para que a gravação de áudio seja realizada.

6. Capacidade de armazenamento
    a. Certifique-se de que o dispositivo tenha espaço de armazenamento suficiente para instalar a aplicação.
7. Autenticação e segurança
    a. A aplicação utiliza métodos de autenticação seguros para proteger os dados dos utilizadores.

Deixamos ainda algumas recomendações de utilização para que a nossa aplicação possa ser utilizada com maior taxa de êxito:

- Ambiente de uso: Para obter os melhores resultados na gravação de vídeos e áudios, utilize a aplicação em um ambiente bem iluminado e com pouco ruído de fundo.

4.2.1.3 Requisitos de armazenamento de dados

Para garantir o armazenamento seguro e eficiente de dados biométricos no projeto de autenticação anti *deepfake*, utilizamos o *Firebase* como a nossa principal ferramenta de armazenamento de dados. A estrutura da base de dados é organizada para suportar o armazenamento e recuperação de dados biométricos de maneira eficiente, mantendo a integridade e segurança das informações.

Somente utilizadores autenticados podem aceder aos seus próprios dados biométricos, enquanto administradores têm acesso a todos os dados para fins de monitoramento e manutenção.

Para garantir que os nossos dados são armazenados de forma segura, utilizamos um sistema de encriptação conhecido como *bcrypt*.

O *bcrypt* é um algoritmo de *hash* adaptativo amplamente utilizado para segurança de senhas. Esta ferramenta foi projetada especificamente para resistir a ataques de força-bruta ao aumentar o custo computacional necessário para gerar *hashes* de senhas. A principal característica adaptativa do *bcrypt* permite ajustar a complexidade do algoritmo ao longo do tempo de criptografia. Isso significa que conforme os recursos computacionais aumentam, o *bcrypt* pode ser configurado para exigir mais tempo e poder de processamento para gerar cada hash. Isso torna os ataques de força-bruta impraticáveis, mesmo com hardware avançado, pois o tempo necessário para testar cada possível combinação de senha aumenta exponencialmente.

Além disso, o *bcrypt* utiliza automaticamente o conceito de "*salting*". O "*salting*" é uma prática recomendada em criptografia onde uma *string* aleatória única (conhecida como "*salt*") é concatenada à senha antes da aplicação do algoritmo de *hash*. Isso garante que senhas idênticas resultem em *hashes* diferentes, mesmo que as senhas originais sejam as mesmas. Dessa forma, evita-se o uso de tabelas *rainbow* e outros métodos de pré-computação de hashes, onde *hashes* de senhas comuns são armazenados para comparação rápida.

Tabelas *rainbow* são estruturas de dados utilizadas em técnicas de quebra de senhas que pré-computam *hashes* de senhas comuns para acelerar o processo de descoberta de senhas originais a partir de seus *hashes*. Elas são compostas por pares de valores que relacionam senhas comuns e seus respetivos *hashes*.

A combinação dessas características faz do *bcrypt* uma escolha robusta para proteger senhas em sistemas de segurança. Ele não oferece apenas resistência contra ataques de força-bruta devido à sua adaptabilidade computacional, mas também assegura a unicidade dos *hashes* através do "*salting*", reforçando a segurança geral das credenciais armazenadas.

Na figura 16, podemos verificar a interface da base dados alojada na estrutura *Firebase,* com a informação acerca de cada utilizador encriptada.

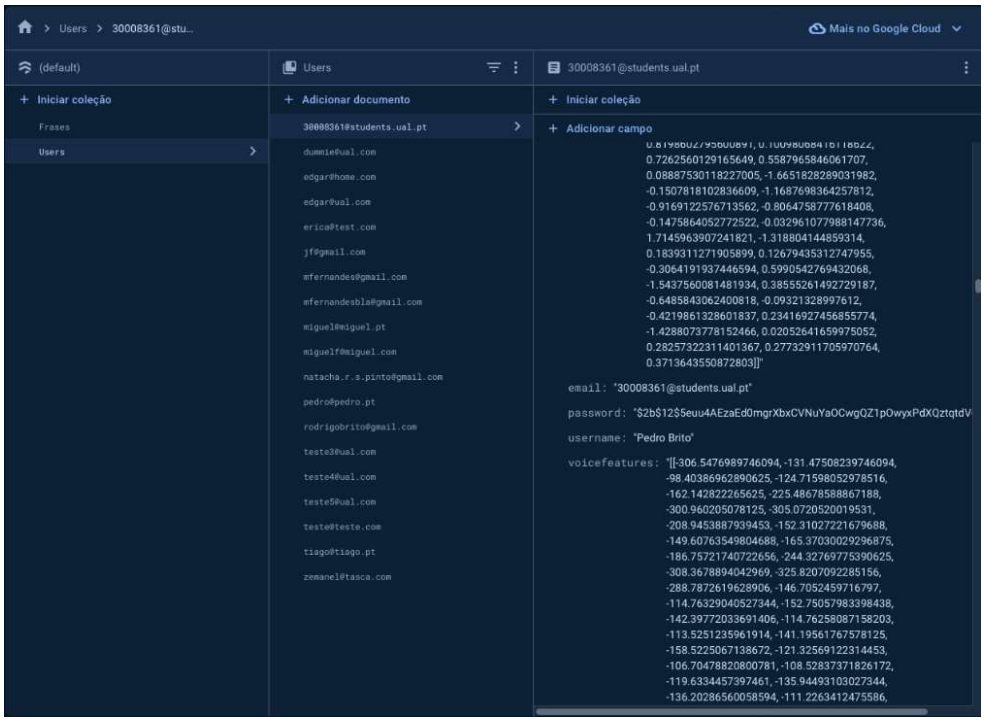

Figura 16 - Interface Firebase

Fonte: Autores

4.2.1.4 Atributos do Aplicativo de software

O nosso sistema de autenticação anti *deepfake* foi criado com o objetivo específico de oferecer autenticação robusta e segura contra-ataques de *deepfake*, utilizando tecnologias avançadas de reconhecimento facial e biometria vocal. Esta abordagem visa garantir a autenticidade e integridade das identidades dos usuários em ambientes digitais sensíveis.

Os atributos do sistema de software propriedades que têm o potencial de afetar o desempenho, a utilização, a eficiência, o quão fácil é usar o sistema, entre outros. Estes atributos permite obter segurança, desempenho, escalabilidade, confiabilidade, manutenibilidade, não repudiação, entre outros. Com estes atributos conseguimos ter bastantes pontos positivos que irão contribuir para o sucesso de uma aplicação, pois irão controlar a forma como a aplicação lida em diferentes circunstâncias e como os utilizadores que a usam pensam sobre ela.

Atributos principais:

- Reconhecimento facial:
    - Captura e comparação de características: Capacidade de capturar imagens faciais em tempo real e comparar com dados biométricos armazenados para autenticação.
    - Deteção de manipulações: Verificação contra deepfakes e alterações fraudulentas nas imagens faciais.
- Biometria de voz:
    - Gravação e analise de padrões vocais: Utilização de algoritmos para capturar e analisar características únicas da voz do usuário para verificação de identidade.
    - Autenticação multifatorial: Combinar reconhecimento facial e biometria vocal para aumentar a segurança da autenticação.
- Integração com *Firebase*:
    - Armazenamento seguro na *cloud*: Utilização do *Firebase* para armazenamento seguro e gestão de dados biométricos, garantindo integridade e confidencialidade.
- Interface intuitiva:
    - Desenho responsivo e amigável: Desenvolvimento de uma interface intuitiva que oferece uma experiência de usuário fluída e eficiente.
    - Manuais de utilizador: A aplicação irá disponibilizar um manual de utilização de forma a orientar os utilizadores na configuração inicial e utilização das funcionalidades de autenticação.

Segurança:

- Criptografia avançada:
    - Algoritmo *Bcrypt*: Utilização do *bcrypt* para a geração de *hashes* de senhas, proporcionando resistência contra-ataques de força-bruta devido à sua adaptabilidade computacional e *salting* automático.

Integrações e extensibilidade:

- APIs: Disponibilização de APIs para integração com outros sistemas e plataformas, permitindo extensibilidade e personalização conforme as necessidades específicas dos clientes e parceiros.

Suporte:

- Suporte técnico: Prestação de suporte técnico eficiente e dedicado para resolver problemas, responder dúvidas e oferecer assistência aos utilizadores da aplicação.

4.2.1.5 Características do ambiente

A combinação de hardware adequado para processamento de vídeo e voz, seja local ou em *cloud*, junto com dispositivos móveis como periféricos para uso do aplicativo, proporciona uma plataforma robusta e acessível para implementação do sistema de autenticação anti *deepfake*. Essa configuração garante não apenas segurança e eficiência na autenticação biométrica, mas também uma experiência de utilizador intuitiva e integrada, adaptada aos padrões de mobilidade e conectividade atuais.

O sistema de autenticação anti *deepfake* foi projetado para operar de forma eficiente e segura em um ambiente híbrido, combinando computação local e em nuvem para garantir desempenho e escalabilidade. As principais características do ambiente são:

- Computação local
    - Servidor local: Utilizamos um servidor local dedicado como nosso servidor principal, equipado com processadores de alta performance e placas gráficas robustas. Este hardware é essencial para lidar com o processamento intensivo necessário para o reconhecimento facial e análise de voz em tempo real.
- Computação em cloud:
    - *Azure speech SDK*: Para o processamento da conversão de voz para texto, utilizamos a solução de computação em *cloud* da "Azure", mais especificamente o serviço *speechsdk* [13]. Este serviço é integrado ao nosso sistema através do *SpeechRecognizer* [14], garantindo uma precisão de 99,9% na conversão de fala em texto.
    - Escalabilidade e confiabilidade: A adoção da Azure para esta fase crítica do processo assegura que o sistema possa lidar com variações na execução do processamento e mantenha disponibilidade contínua, garantindo que a fase de segurança de conversão de voz tenha um número mínimo de falhas.

As vantagens de utilização de um ambiente híbrido são que este, permite combinar a computação local e em *cloud*, permitindo que o processamento intensivo seja realizado localmente enquanto tarefas específicas, como a conversão de voz para texto, sejam executadas na nuvem.

Quanto aos periféricos utilizados, os utilizadores têm a necessidade de possuir um dispositivo móvel, pois este será essencial para a recolha dos dados biométricos.

- *Smartphones* e/ou *Tablets*:
    - Equipados com câmeras frontais para captura de imagens faciais.
    - Microfones para gravação de amostras de voz.
    - Conexão à internet para comunicação eficiente com o servidor local e serviços em *cloud*, permitindo que os dados sejam enviados e recebidos em tempo real.

4.2.2 Componentes da aplicação

O sistema de autenticação anti deepfake foi desenvolvido por meio da aplicação e integração de diversas bibliotecas especializadas, utilizando uma variedade de ferramentas disponibilizadas. Para garantir a eficácia e segurança do sistema, o nosso servidor local foi configurado com as seguintes bibliotecas:

Server *Python*

- Manipulação de *strings* e arquivos
    - '*Shutil*' - Biblioteca para operações de manipulação de arquivos e diretórios.
    - '*Flask*' – Estrutura web para python, facilita a criação de APIs e aplicações web.
    - '*Firebase_admin*' – API para interagir com os serviços do Firebase, faz a gestão da autenticação dos utilizadores, base de dados e armazenamento.
    - '*Import re'* - Utilizada para pesquisar e manipular strings através de expressões regulares, facilitando a validação e transformação de dados textuais.

- ‘*Import os’* - Fornece uma interface para manipulação de arquivos, pastas e diretórios no sistema operacional, permitindo a leitura, escrita e organização de dados de forma programática.
- *‘secure_filename*’ - é uma função fornecida pelo módulo ‘*werkzeug.utils’* da estrutura ‘*Flask*’ que é usada para proteger e tornar os nomes de ficheiros utilizáveis antes de os armazenar no servidor. Quando os utilizadores fazem upload de ficheiros, os nomes desses ficheiros podem conter caracteres especiais, espaços ou outros elementos potencialmente perigosos. Ao usar *secure_filename*, garante-se que o nome do ficheiro é transformado numa versão mais segura. A função remove ou substitui caracteres que poderiam ser usados para fins maliciosos. [14]

- Criptografia
  - ‘*bcrypt*’ – Posteriormente mencionado como a estrutura de criptografia de senhas. Gera *hashes* seguros utilizando *salting* automático e um algoritmo de *hashing* adaptativo, dificultando ataques de força bruta.
- Manipulação de dados
  - ‘*json*’ - Utilizada para manipulação de dados no formato JSON, permitindo a leitura, escrita e transformação de dados estruturados de forma eficiente.
  - ‘*pprint*’ – Biblioteca utilizada para impressão de estruturas de dados complexas de forma legível e organizada, facilitando a depuração e análise de dados.

Cada uma destas bibliotecas desempenha um papel crucial na arquitetura do sistema, garantindo que todas as etapas do processo de autenticação, desde a captura e processamento de dados até a verificação e armazenamento seguro, sejam realizadas com precisão e segurança.

4.2.2.1 Back-end

No *Back-end* as soluções utilizadas para o processamento de dados de reconhecimento facial e biometria de voz, foram todas implementadas no servidor local na linguagem *python*, que temos à nossa disponibilidade, fazendo uso das diversas bibliotecas e tecnologias disponíveis. As bibliotecas utilizadas foram as seguintes:

- Processamento numérico e analise de dados
    - '*Numpy*' – O '*numpy*' é uma biblioteca fundamental para computação numérica, oferecendo suporte para *arrays* e vetores multidimensionais, além de uma ampla gama de operações matemáticas.
    - '*SciPy'* – O módulo '*scipy.spatial.distance*' faz parte da biblioteca *SciPy* em *Python* e fornece um conjunto abrangente de funções para calcular distâncias entre pontos ou conjuntos de pontos em vários espaços. Este módulo é comumente usado em campos como *machine learning*, análise de dados e geometria computacional para tarefas que envolvem a medição de similaridade ou dissimilaridade entre pontos de dados.
        - Esta biblioteca contém uma variedade de métricas de distância, incluindo:
            - Distância Euclidiana: A distância em linha reta entre dois pontos no espaço euclidiano.
            - Distância de *Manhattan*: A soma das diferenças absolutas das coordenadas.
            - Distância de *Chebyshev*: A diferença absoluta máxima ao longo de qualquer dimensão de coordenada
            - Distância de *Minkowski*: Uma generalização das distâncias Euclidiana e *Manhattan*, parametrizada por um parâmetro de potência p.
            - Distância Cosseno: Mede o cosseno do ângulo entre dois vetores não nulos, útil para análise de texto e outras aplicações onde a direção é mais importante que a magnitude.
            - Distância de *Jaccard*: Mede a dissimilaridade entre conjuntos de amostras, útil em cenários de dados binários.
            - Distância de *Hamming:* A proporção de bits que diferem entre dois vetores binários.
        - Cálculo de distâncias de pares em observações num espaço n-dimensional:
            - '*pdist*' - Calcula distâncias de pares entre pontos em um único *array*.

- '*cdist*' - Calcula distâncias entre cada par de pontos de dois *arrays* diferentes.

- Matrizes de distância:
  - Utilização de funções para trabalhar com matrizes de distância, como '*squareform*' que converte entre formatos de vetor de distância e formatos de matriz quadrada.

- '*scipy.spatial.distance.euclidean*' - Específica para o cálculo da distância euclidiana entre dois pontos, usada em várias análises de dados espaciais.
- '*fastdtw*': Implementa o algoritmo *Dynamic Time Warping*, usado para medir a similaridade entre duas sequências temporais, como amostras de voz. [17]
- '*Tensorflow*' – Estrutura utilizada para construção e treino de modelos de inteligência artificial.

- Processamento de áudio
  - '*librosa*' - Biblioteca especializada em processamento e análise de áudio, utilizada para extrair características e manipular amostras de áudio, como por exemplo, extração de características de áudio como MFCCs *(Mel-frequency cepstral coefficients*), análise de frequência e tempo das amostras de voz.
    - Esta biblioteca é muito poderosa em termos de extração e manipulação de amostras de áudio a metodologia desta biblioteca baseia-se no carregamento de um sinal de áudio, na extração das características através dos parâmetros abaixo mencionados e depois na sua concatenação:
      - MFCC *(Mel-frequency cepstral coefficients*) – Os MFCC capturam as propriedades do espectro de curto prazo de um sinal de áudio. Podemos ver na imagem abaixo um exemplo de captura MFCC

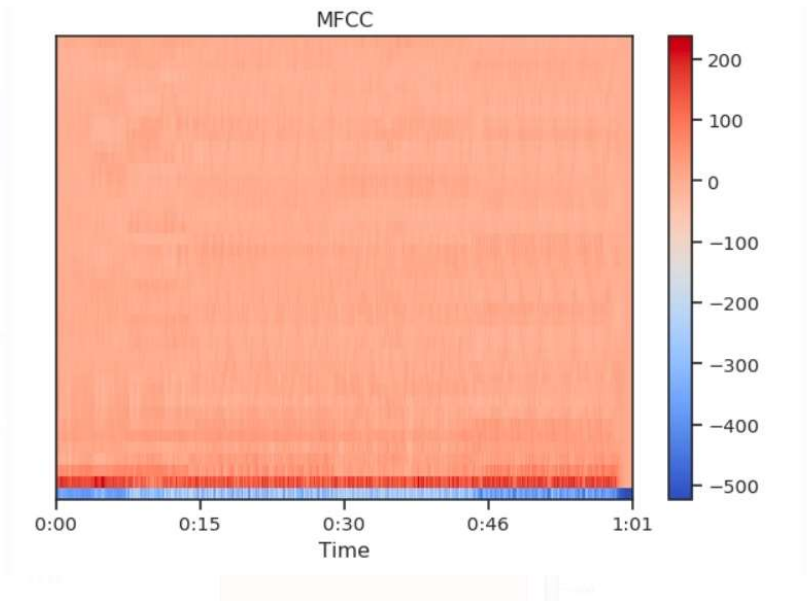

Figura 17 - MFCC

Fonte: https://librosa.org/doc/main/generated/librosa.feature.mfcc.html

- *'Chromagram'* – O *chromagram* representa a energia projetada em cada uma das 12 notas da escala cromática, tal como podemos observar na figura 17.

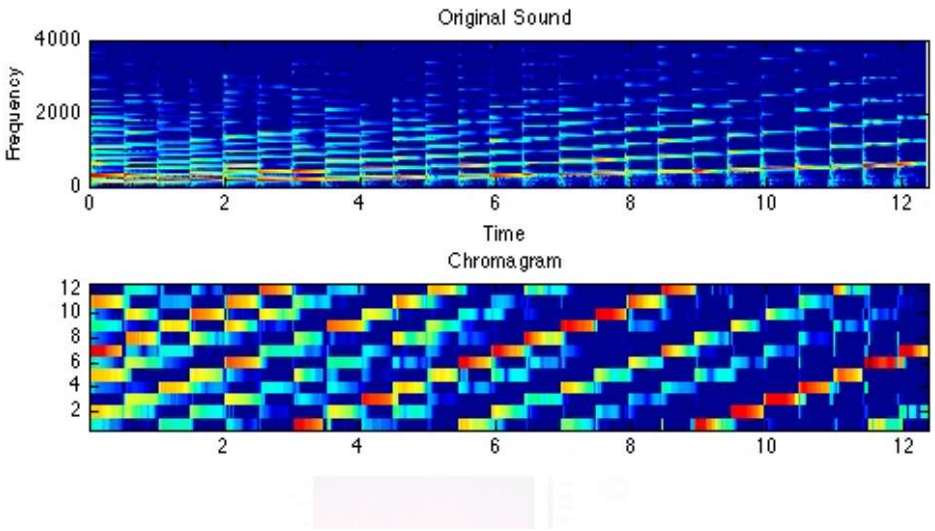

Figura 18 - Chromagram

Fonte: https://www.researchgate.net/figure/The-spectrogram-top-and-chromagram-bottom-of-an-ascending-scale_fig3_314918556

- Espectrograma Mel ou (*Mel Spectogram*) - Extrai a espectrograma Mel, que é uma representação do espectro de frequências do áudio utilizando a escala de frequência Mel. É utilizado em aplicações de reconhecimento de fala e análise musical devido à sua representação mais próxima da perceção humana de som.

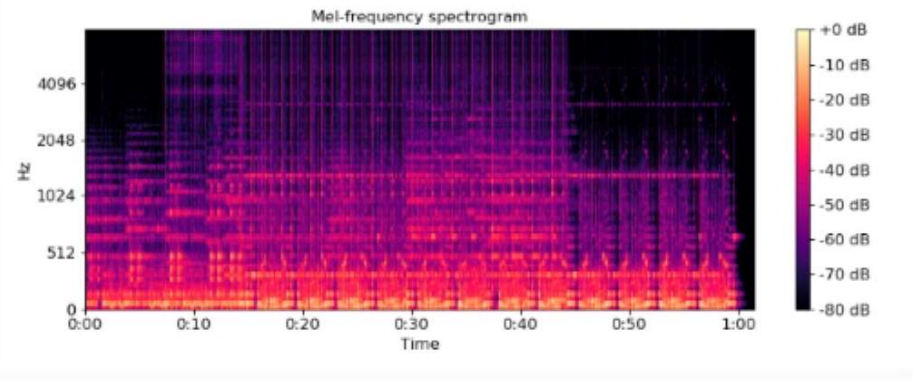

Figura 19 - Mel Spectogram

Fonte: https://librosa.org/doc/main/generated/librosa.feature.mfcc.html

- Contraste de espectros (*Spectral Contrast*) - Extrai o contraste espectral do sinal de áudio, medindo a diferença entre picos e vales no espectro. É bastante útil para distinguir entre diferentes tipos de sons e texturas musicais, sendo relevante em tarefas de classificação de áudio.

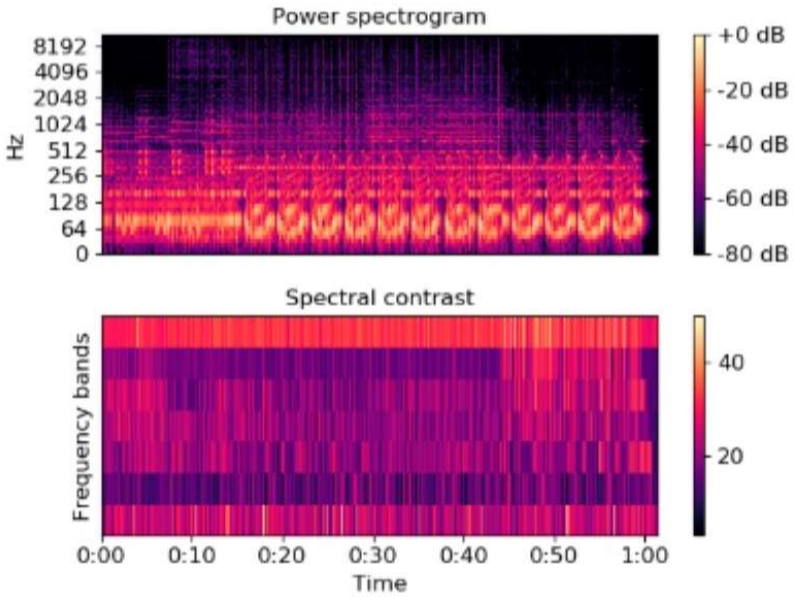

Figura 20 - Spectral Contrast

Fonte: https://librosa.org/doc-playground/0.7.2/generated/librosa.effects.preemphasis.html

- Por último, apos a extração de todas as características acima mencionadas elas são concatenadas em um único *array* de características composta, que pode ser usada para tarefas de reconhecimento de padrões, classificação e outras análises no domínio do áudio.

  - '*azure.cognitiveservices speech'*: Utilizamos uma parte dos serviços de *cloud* da Azure, para a conversão de fala em texto, como parte de uma dos fatores de autenticação, tornando-se assim imprescindível para a biometria de voz do sistema.

- Reconhecimento facial:
  - '*moviepy.editor*' - Biblioteca para edição e manipulação de vídeos em Python, permitindo cortes, concatenações e extrações de *frames*. Trabalha como processador dos vídeos para analise de autenticidade.
  - '*cv2' (OpenCV)* – Biblioteca de visão computacional, trabalha no processamento de imagens e vídeos, incluindo reconhecimento facial.

- ‘*dlib*’ - Biblioteca de *machine learning* com ferramentas para análise de dados e visão computacional. Esta biblioteca complementa o *OpenCV* em tarefas avançadas de processamento de imagens.
- ‘*imutils.face_utils’* - Facilita o processamento de imagens faciais com OpenCV, auxilia na extração e manipulação de características faciais.
- ‘*Deepface’* - Biblioteca para reconhecimento facial utilizando aprendizagem profunda, com a finalidade de identificação de faces e prova de vida (verificação facial e deteção de veracidade).
-

Todas estas bibliotecas são integradas para suportar as funcionalidades do sistema de autenticação anti *deepfake*, abrangendo desde o processamento de imagens e vídeos até a análise de áudio e a integração com serviços em *cloud* e base de dados.

4.2.2.2 Front-end

No *Front-end* a solução escolhida foi o *Flutter* porque ser um *framework* de estrutura de código aberto da Google para o desenvolvimento de aplicativos móveis, web e desktop, conhecido por permitir o desenvolvimento de aplicações multiplataforma com uma única base de código tendo como principais características:

Figura 21 - Logotipo Flutter

Fonte: https://flutter.dev

Linguagem de Programação Dart: É uma linguagem também ela desenvolvida pela Google, orientada a objetos com uma sintaxe similar ao *JavaScript;*

Figura 22 - Logotipo Dart

Fonte: https://dart.dev

UI Nativa e Rápida: Permite criar interfaces de utilizador que são visualmente atraentes e com uma performance muito elevada, usando o seu próprio motor de renderização (*Skia*), o que permite um enorme controle sobre cada pixel;

***Widgets* Personalizáveis:** Oferece uma vasta coleção de *widgets* pré-construídos (blocos de construção dos aplicativos) que podem ser facilmente personalizados;

Desenvolvimento Rápido: Com o recurso de "*hot reload*", durante o desenvolvimento é possível ver as mudanças no código refletidas em tempo real no aplicativo em execução, sem a necessidade de estar sempre a recompilar;

Código Único para Várias Plataformas: É possível escrever um único código base que funciona em Android, iOS, web e até mesmo em desktop, reduzindo significativamente o esforço de desenvolvimento, figura 23.

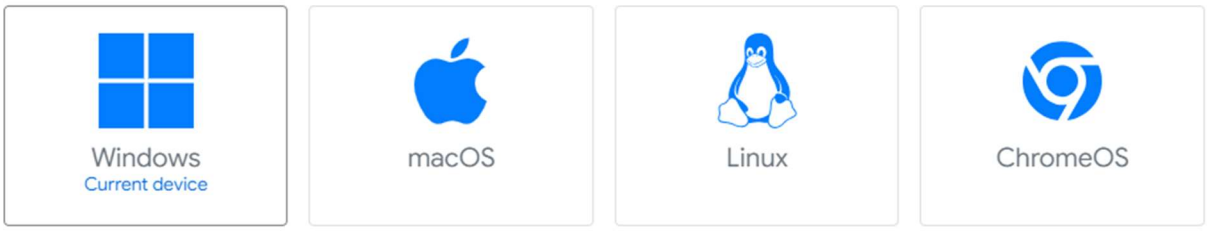

Figura 23 - Plataformas em Código Único

Utilizamos vários componentes e funcionalidades desta linguagem nativa tais como:

- '*main.dart*' – Inicialização da aplicação define o ponto de entrada da aplicação *Flutter* e configurações iniciais, como tema e caminho inicial.

- ‘*my_drawer_header.dart*’ – Interface de login, providencia elementos visuais para o drawer (menu lateral) da aplicação, que pode incluir opções de login e informações do utilizador
- *‘bio/bio_reg_create.dart’* – Interface de criação de registo biométrico, permite ao utilizador criar seu registo biométrico, capturando e armazenando dados biométricos como voz e video.
- ‘*bio/bio_reg_welcome.dart’* - Orienta o usuário sobre o processo de captura e registo de dados biométricos, garantindo qualidade e precisão na recolha.
- *‘screens/deletebio.dart’* – Interface que permite eliminar o registo biométrico do usuário e criar um novo. Fornece opções para o utilizador fazer a gestão dos seus dados biométricos, incluindo a exclusão e recriação do registo para manter a segurança e precisão do sistema.
- *‘screens/login.dart’* – Faz a gestão do fluxo de autenticação, permitindo que o utilizador inicie ou encerre sessões de forma segura, além de redirecionar para a página inicial após login bem-sucedido.
- *‘screens/splash.dart’* - Proporciona uma experiência visual atraente ao carregar a aplicação, exibindo o logotipo e preparando o ambiente para o utilizador.
- *‘screens/Welcome.dart’* - Celebra a autenticação bem-sucedida, fornecendo uma mensagem personalizada de boas-vindas e possivelmente orientações adicionais sobre como proceder.
- *‘services/account_status.dart’* - Avalia se o utilizador possui um registo biométrico válido e direciona para o ecrã apropriado com base no estado desse registo.
- *‘services/api.dart’* – Conexão com a API, facilita a comunicação entre a aplicação e servidores externos, permitindo o envio e recebimento de dados relacionados à autenticação e gestão de utilizadores.
- *‘utils/constants.dart’* - Armazena valores fixos como *URLs* de API, chaves de autenticação e outras constantes necessárias em vários pontos da aplicação.

Cada um destes arquivos desempenha um papel crucial no seu sistema de autenticação anti *deepfake*, garantindo funcionalidades robustas e seguras para autenticação biométrica e gestão de utilizadores. Esta estrutura ajuda a organizar o desenvolvimento e facilita a manutenção da aplicação ao longo do tempo.

5 Testes e Avaliação

Para validar a arquitetura do sistema, realizamos diversos testes levando em conta diferentes condições, como variações de voz, número de pessoas em frente do equipamento e o comportamento dos olhos (abertos ou fechados).

Os testes realizados foram:

- Capturar um individuo piscando os olhos e mexendo a boca (leitura).
- Capturar um individuo a uma distância de enquadramento de foto tipo passe.
- Capturar um individuo a efetuar a leitura de uma frase pré-atribuída.
- Capturar vários indivíduos na mesma imagem (erro).

Os fatores que mais afetaram os resultados foram:

A falta de número de indivíduos para o reconhecimento, o que reduziu significativamente a precisão de validação, muitas vezes resultando em falhas no reconhecimento facial e de voz.

Com esses testes, conseguimos identificar pontos críticos que impactam a eficácia do sistema e estamos comprometidos em melhorar e otimizar nosso projeto para melhorar ainda mais a segurança e a usabilidade da autenticação.

5.1 Teste e Avaliação do Reconhecimento Facial

A avaliação do sistema de reconhecimento facial é essencial para garantir a precisão, robustez e segurança do sistema de autenticação anti *deepfake*.

Foram executados vários testes que serão descritos abaixo para que fosse possível avaliar o desempenho do nosso modelo de reconhecimento facial. Para os testes realizados e a análise dos resultados obtidos baseamo-nos na comparação de imagens dos registos previamente guardados na nossa base de dados e também aos mecanismos utilizados na prova de vida, como deteção de movimento facial, piscar de olhos, abertura e fecho de boca.

- Metodologia

- Utilizamos um conjunto diversificado de imagens e vídeos faciais, incluindo diferentes idades, gêneros, etnias e condições de iluminação, para garantir que o modelo seja treinado e testado em um amplo espectro de casos reais.
- Os testes foram conduzidos em um servidor local equipado com um processador de alto desempenho
- Optamos por utilizar a biblioteca *DeepFace* [9] para o reconhecimento facial e técnicas de prova de vida (*liveness*) e outras ferramentas auxiliares tais como o *OpenCV* e *Dlib* para processamento de imagens e vídeos, além de técnicas de aprendizagem profunda com o *Tensorflow.*

- Procedimentos de teste
  - Autenticação – Testamos o sistema com múltiplas tentativas de autenticação usando imagens reais dos usuários, vídeos e *deepfakes*
  - Prova de vida - Implementamos testes de prova de vida que requerem movimentos faciais específicos, como piscar de olhos e abrir e fechar a boca, para garantir que a pessoa na frente da câmera seja real e não uma imagem estática, foto ou vídeo manipulado.
  - Desempenho - Medimos o tempo de processamento para cada autenticação, avaliando a eficiência do sistema em condições de uso real.
  - Precisão - Calculamos a taxa de falsos positivos (falsas aceitações) e falsos negativos (falsas rejeições) para avaliar a precisão do modelo.
- Avaliação
  - Vantagens
    - Precisão – Uma taxa de precisão considerável e baixa taxa de erros (falsos positivos e falsos negativos) garantem a confiabilidade do sistema.
    - Resistência a *deepfakes* – A capacidade do sistema de identificar deepfakes minimiza o risco de ataques de falsificação.
    - Eficiência - O rápido tempo de processamento assegura que o sistema possa ser utilizado em tempo real sem comprometer a experiência do utilizador.

- Prova de vida - A implementação de testes de prova de vida adiciona uma camada adicional de segurança, garantindo que a pessoa autenticada seja real.

- Pontos de melhoria
  - Condições de iluminação - Embora o sistema tenha tido um bom desempenho em diversas condições de iluminação, algumas variações podem ainda afetar a precisão.
  - Diversidade de dados - Um continuo treino de dados e testes é necessário para assegurar que o sistema se mantenha o mais atualizado, seguro e robusto possível.
  - Prova de vida – Aumentar a complexidade dos testes de prova de vida para incluir vários cenários onde existam diferentes condições de ambiente para garantir de que a prova de vida é fidedigna.
  -

Figura 24 - Teste Reconhecimento Facial

Fonte: Autores

Os testes e avaliações realizados demonstram que o sistema de reconhecimento facial desenvolvido é eficaz para autenticação anti *deepfake*, combinando precisão, eficiência e robustez. A inclusão de mecanismos de prova de vida, como deteção de movimento facial e análise em tempo real, adiciona uma camada crítica de segurança, assegurando que apenas utilizadores reais possam ser autenticados. A adoção de técnicas de processamento de imagens e aprendizagem profunda, aliada a um ambiente de teste rigoroso, assegura que o sistema possa ser confiável em aplicações práticas.

5.2 Teste e Avaliação da Biometria de Voz

Para a avaliação da biometria de voz, é essencial considerar os diversos componentes utilizados na implementação desta solução. O objetivo é garantir tanto a precisão quanto a segurança do sistema. Nesse contexto, focamos principalmente nas tecnologias fornecidas pela biblioteca *librosa* e na tecnologia de reconhecimento de voz para texto da Azure.

Os testes foram conduzidos para assegurar que a extração de características de áudio e a conversão de voz em texto fossem realizadas com precisão. A biblioteca *librosa* foi utilizada para extrair características acústicas detalhadas do áudio, como MFCCs, *chromagram* e *Mel Spectogram* e *contrast*. Essas características são fundamentais para a análise e reconhecimento da biometria de voz.

Além disso, a tecnologia de reconhecimento de voz para texto da Azure foi integrada para converter amostras de voz em texto com precisão. Esta combinação de tecnologias permitiu a criação de um sistema robusto que, não só autêntica os utilizadores com base em suas caraterísticas vocais únicas, mas também garante a segurança contra tentativas de fraude e *deepfakes.*

- Metodologia
    - Conjunto de áudios - Utilizamos um conjunto diversificado de gravações de voz, incluindo diferentes idades, gêneros, etnias e condições de ruído, para garantir que o modelo seja treinado e testado em um amplo espectro de casos reais.

- Os testes formam conduzidos em um servidor local onde utilizamos um componente da *Azure Cognitive services* para a conversão de fala em texto e a biblioteca *librosa* para a extração de características vocais, como MFCC (*Mel-Frequency Cepstral Coefficients*), *Contrast*, *Mel Spectrogram*, e *Chromagram*, e DTW (*Dynamic Time Warping*) para medir a similaridade das sequências vocais. [17]

- Procedimentos de teste
  - Autenticação – Testamos o sistema com múltiplas tentativas de autenticação usando gravações de voz reais de utilizadores, áudios manipulados e *deepfakes*
  - Desempenho – Medimos o tempo de processamento para cada autenticação, avaliando a eficiência do sistema em condições de uso real
  - Precisão - Calculamos a taxa de falsos positivos e falsos negativos para avaliar a precisão do modelo.
  - Prova de vida - Implementamos testes de prova de vida que requerem a pronúncia de frases específicas para garantir que a pessoa que fala seja real e não uma gravação. Para que a prova de vida seja fidedigna, utilizamos diversos parâmetros tais como mencionados abaixo:
    - Tecnologia *Azure Speech* – Utilizamos o serviço *SpeechRecognizer,* para que seja possível obter uma taxa de precisão bastante elevada na conversão de fala em texto, assegurando que a frase pronunciada pela o utilizador seja corretamente identificado.
    - Analise de características vocais - A biblioteca *librosa* foi utilizada para extrair características vocais como MFCC, *Contrast*, *Mel Spectrogram*, e *Chromagram*, que foram fundamentais para diferenciar entre vozes reais e *deepfakes*.
      - MFCC (*Mel-Frequency Cepstral Coefficients)* - Captura as propriedades do timbre da voz, essencial para a identificação do locutor.
      - *Contrast* - Realça as diferenças entre regiões de alta e baixa energia em uma espectrograma, útil para distinguir características únicas da voz.

- *Mel Spectogram* - Representa a distribuição de energia em diferentes frequências, fornecendo uma visão detalhada das características acústicas.
- *Chromagram* - Capta a intensidade das diferentes notas musicais, ajudando a identificar padrões específicos da voz.
- DTW *(Dynamic Time Warping)* - Utilizado para medir a similaridade entre as sequências vocais extraídas, o DTW foi essencial para comparar a fala do utilizador em tempo real com os padrões de voz armazenados. A análise DTW permitiu ajustar as variações temporais e garantir uma correspondência precisa, resultando em uma taxa de sucesso elevada na autenticação. [17]

- Avaliação
  - Vantagens
    - Precisão - A alta taxa de precisão e baixa taxa de erros (falsos positivos e negativos) garantem a confiabilidade do sistema.
    - Resistência - A capacidade do sistema de identificar *deepfakes* minimiza o risco de ataques de falsificação.
    - Eficiência - O rápido tempo de processamento assegura que o sistema possa ser utilizado em tempo real sem comprometer a experiência do utilizador.
  - Pontos de melhoria
    - Ruido de fundo - Ambientes com alto ruído de fundo ainda representam um desafio, embora o uso de MFCC e outras técnicas de extração de características ajude a mitigar esses efeitos
    - Variações na pronúncia - Diferenças significativas na pronúncia ou mudanças na voz do utilizador (devido a saúde ou outras condições) podem afetar a precisão do sistema, exigindo ajustes contínuos e treinamento adicional.

5.3 Teste e Avaliação da Autenticação Integrada anti *Deepfake*

O sistema de autenticação integrado *anti-deepfake*, que combina reconhecimento facial e biometria de voz, foi rigorosamente testado e avaliado para garantir sua eficácia e segurança. Para testar a integração do sistema de autenticação multifatorial, foi necessário combinar todas as fases envolvidas na criação desta aplicação.

O nosso grupo optou por criar um ambiente combinado, onde a autenticação só é concluída com sucesso se o utilizador passar por múltiplos fatores de verificação. O processo de autenticação facial envolve a gravação de um vídeo, que é então cortado em *frames* para comparação com os dados biométricos previamente armazenados em nossa base de dados. Durante este processo, ocorre a deteção de prova de vida, verificando a presença de movimentos faciais, abrir e fechar a boca e os olhos, para garantir que a imagem capturada seja de uma pessoa viva e não de uma fraude.

Simultaneamente, a autenticação da biometria de voz é realizada. O sistema verifica se a frase exibida no ecrã é dita corretamente pelo utilizador e reconhecida pelo sistema através da tecnologia Azure. A voz do utilizador é comparada com os dados armazenados na base de dados, utilizando todas as ferramentas fornecidas pela biblioteca *librosa*, conforme mencionado anteriormente.

Esta abordagem integrada assegura uma autenticação robusta e segura, combinando reconhecimento facial e de voz para criar um sistema de autenticação multifatorial resistente a ataques de *deepfake* e outras tentativas de fraude.

Figura 25 - Autenticação bem-sucedida

Fonte: Autores

Resultados e Discussão

Os testes de integração desempenharam um papel crucial no desenvolvimento do nosso sistema de autenticação. Ao focar na interação sistemática entre o *back-end* e o *front-end*, conseguimos criar uma aplicação mais robusta, eficiente e confiável.

Resultados dos Testes de Integração:

Os testes de integração permitiram identificar e resolver uma série de problemas que poderiam ter comprometido a funcionalidade e a usabilidade da aplicação. Alguns dos principais benefícios obtidos foram:

- Melhoria na Comunicação: A validação das requisições garantiu que o *front-end* e o *back-end* comunicassem de maneira eficiente e precisa, reduzindo a incidência de erros e mal-entendidos entre os módulos.
- Resiliência do Sistema: O tratamento de erros robusto aumentou a resiliência da aplicação, permitindo que ela lidasse de maneira elegante com falhas e problemas de comunicação.
- Eficiência no Desenvolvimento: A implementação de testes automatizados e o uso de *logs* e monitorização facilitaram o trabalho dos programadores, permitindo uma identificação rápida dos problemas e garantindo a integridade das novas funcionalidades implementadas.

```
-> RECOGNIZE_VOICE: Falha na autenticação: Voz não reconhecida! Distância:155374.16854170663
-> LIVENESS: Boca movimentos: 0
-> LIVENESS: Motion valor: 0.12451171875
-> VALIDATE_USER: Face:False Frase:False Voz:False Liveness:False
89.115.33.202 - - [27/Jun/2024 18:20:16] "POST /validate_user HTTP/1.1" 200 -
89.115.33.202 - - [27/Jun/2024 18:20:29] "POST /camera_save HTTP/1.1" 200 -
-> FACE_DETECT: # Autenticações:10 / # Similaridades:200
-> FACE_DETECT: Utilizador autorizado! Fotos:10 Similaridades:200 80%
MoviePy - Writing audio in
MoviePy - Done.
-> RECOGNIZE_SPEECH: Frase Original: o microondas está avariado
-> RECOGNIZE_SPEECH: Frase Reconhecida: o microondas está avariado
MoviePy - Writing audio in
MoviePy - Done.
-> RECOGNIZE_VOICE: Autenticado com sucesso! Distância:41401.92427820719
-> LIVENESS: Boca movimentos: 86
-> LIVENESS: Motion valor: 2.79072265625
-> VALIDATE_USER: Face:True Frase:True Voz:True Liveness:True
89.115.33.202 - - [27/Jun/2024 18:20:35] "POST /validate_user HTTP/1.1" 200 -
89.115.33.202 - - [27/Jun/2024 18:20:46] "POST /userinfo HTTP/1.1" 200 -
89.115.33.202 - - [27/Jun/2024 18:20:58] "POST /userinfo HTTP/1.1" 200 -
-> PHRASE: O trabalho está concluído.
89.115.33.202 - - [27/Jun/2024 18:21:13] "POST /phrase HTTP/1.1" 200 -
89.115.33.202 - - [27/Jun/2024 18:21:21] "POST /camera_save HTTP/1.1" 200 -
-> FACE_REGISTER: Reconhecimento: 94%
MoviePy - Writing audio in
MoviePy - Done.
-> RECOGNIZE_SPEECH: Frase Original: o trabalho está concluído
-> RECOGNIZE_SPEECH: Frase Reconhecida: o trabalho está concluído
MoviePy - Writing audio in
MoviePy - Done.
-> VOICE_REGISTER: Utilizador registado na Base de Dados.
-> LIVENESS: Boca movimentos: 83
-> LIVENESS: Motion valor: 2.02041015625
-> REGISTER_USER: Face:True Frase:True Liveness:True
89.115.33.202 - - [27/Jun/2024 18:21:26] "POST /register_user HTTP/1.1" 200 -
-> PHRASE: A cidade é grande.
89.115.33.202 - - [27/Jun/2024 18:21:29] "POST /phrase HTTP/1.1" 200 -
89.115.33.202 - - [27/Jun/2024 18:21:41] "POST /camera_save HTTP/1.1" 200 -
-> FACE_DETECT: # Autenticações:10 / # Similaridades:250
-> FACE_DETECT: Utilizador autorizado! Fotos:10 Similaridades:250 100%
MoviePy - Writing audio in
MoviePy - Done.
-> RECOGNIZE_SPEECH: Frase Original: a cidade é grande
-> RECOGNIZE_SPEECH: Frase Reconhecida: a cidade é grande
MoviePy - Writing audio in
MoviePy - Done.
-> RECOGNIZE_VOICE: Autenticado com sucesso! Distância:21881.2919406635
```

Figura 26 - Resultados servidor

Fonte: Autores

6 Conclusões

O desenvolvimento do Sistema de Autenticação Anti *Deepfake*, um projeto escolhido unanimemente pelo nosso grupo de trabalho, demonstrou ser não apenas pertinente, mas também de extrema importância no contexto atual, marcado por ameaças significativas de manipulações digitais sofisticadas. *Deepfakes*, criados através de técnicas avançadas de aprendizagem profunda como Redes Generativas Adversariais (GANs) e transferência de estilo, têm consequências graves em áreas como desinformação, manipulação política, danos à reputação, fraudes e invasão de privacidade, com especial destaque para o setor bancário.

Diante dessas ameaças, o nosso projeto visou desenvolver uma solução robusta de autenticação que integrasse múltiplos fatores de segurança. A escolha de um sistema de autenticação multifatorial, englobando email e password, reconhecimento facial, biometria de voz, verificação de frase e prova de vida, revelou-se essencial para garantir um alto nível de proteção. Optamos por uma abordagem onde todos os fatores de autenticação são processados simultaneamente, permitindo um registo único e contínuo para o utilizador, simplificando a experiência e aumentando a segurança.

Tecnologicamente, a implementação do projeto utilizou a linguagem de programação *Python* para centralizar o processamento de dados, integrando-se com uma base de dados (*Firebase*) e uma interface de usuário desenvolvida em *Flutter*. No nosso projeto a autenticação por reconhecimento facial, utilizamos *Flutter, Google ML Kit, TensorFlow Lite* e o *FaceNet512,* foi desenvolvido para melhorar a segurança e a experiência do usuário. Focamos em quatro aspetos principais: o sistema de reconhecimento facial, a implementação de uma solução de prova de vida, o sistema de reconhecimento de voz e ainda o reconhecimento de uma frase.

Durante o desenvolvimento, enfrentamos desafios substanciais, especialmente na autenticação por voz, que exigiu uma pesquisa aprofundada e ajustes técnicos para alcançar uma precisão e credibilidade equiparáveis ao reconhecimento facial.

Após realizar extensivos testes e refinamentos, aprimoramos nosso banco de dados e algoritmos de comparação, garantindo a máxima precisão e veracidade nas validações. O

trabalho árduo e colaborativo de pesquisa, aliado à orientação docente e ao desenvolvimento contínuo, culminou em uma aplicação robusta e funcional. O nosso sistema foi capaz de oferecer uma autenticação segura e confiável, mitigando eficazmente os riscos associados aos *deepfakes*.

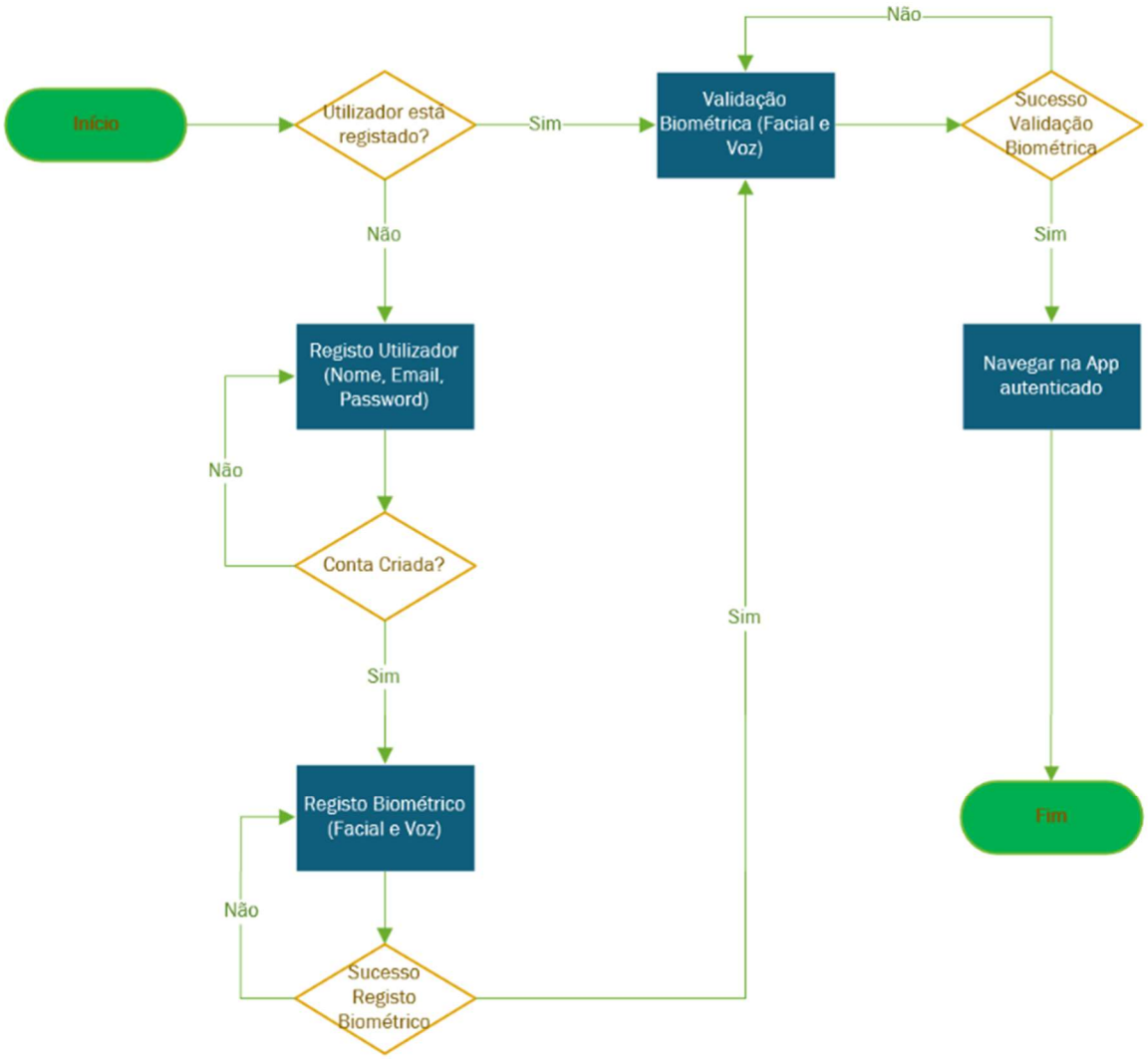

Figura 27 - Fluxo do processo

Fonte: Autores

Em suma, a conclusão deste projeto não só valida a viabilidade técnica de sistemas de autenticação avançados, mas também sublinha a importância crucial de tais tecnologias na proteção contra ameaças digitais contemporâneas. Este projeto reforça a necessidade de segurança cibernética robusta e a prevenção de fraudes, demonstrando o impacto positivo que soluções inovadoras podem ter na sociedade. Este sistema de autenticação anti *deepfake*

representa um passo significativo em direção a uma proteção mais eficaz e confiável contra as manipulações digitais, contribuindo para um ambiente digital mais seguro e confiável.

Referências

[1] Dominic Forrest. Challenges in voice biometrics: Vulnerabilities in the age of *deepfakes*, 2024, February 15, Iproov. https://bankingjournal.aba.com/2024/02/challenges-in-voice-biometrics-vulnerabilities-in-the-age-of-*deepfakes*/

[2] Marcelo Peixoto. A era dos *Deepfakes*: como a biometria de voz é uma ferramenta crucial na prevenção de fraudes, 2023, Julho 4, TI Inside. https://pt.linkedin.com/pulse/era-dos-*deepfakes*-como-biometria-de-voz-%C3%A9-uma-ferramenta-crucial

[3] Bianca Gonzalez. Cybercriminals use malware to obtain face biometrics, break into banking apps, 2024, February 15, Infosec Institute. https://www.biometricupdate.com/202402/cybercriminals-use-malware-to-obtain-face-biometrics-break-into-banking-apps

[4] Šandor, Oskar — "Resilience of biometric authentication of voice assistants against *deepfakes*". BRNO, 2023. Bachelor's Thesis, presented to BRNO UNIVERSITY OF TECHNOLOGY. https://theses.cz/id/nr5tm7/OS_BP_FINAL.pdf

[5] REŠ, Jakob — "Testing the robustness of a voice biometrics system against *deepfakes*". BRNO, 2023. Master's Thesis, presented to BRNO UNIVERSITY OF TECHNOLOGY. Available in: https://theses.cz/id/wfbqy4/DP.pdf

[6] https://www.redhat.com/pt-br/topics/api/what-are-application-programming-interfaces [Acedido em 19 abril 2024]

[7] https://github.com/parvatijay2901/FaceNet_FR

[8] https://viso.ai/computer-vision/deepface/

[9] https://pypi.org/project/deepface/

[10] Abreu, Viviana Rubina Gonçalves - Reconhecimento Facial - Comparação do Uso de Descritores Geométricos Heurísticos e

Aprendizagem Profunda. Coimbra fevereiro de 2021, Tese de Mestrado na Universidade de Coimbra.
Disponível em:
https://hdl.handle.net/10316/94339

[11] Rout, Siddharth - maio 2019, Tese de Bacharelato em Tecnologia na Indian Institute of Technology Madras, Chennai
Disponível em:
https://arxiv.org/pdf/2111.02987

[12] Chiţu, A. G., Rothkrantz, L. J., Wiggers, P., & Wojdel, J. C. (2007). Comparison between different feature extraction techniques for audio-visual speech recognition. Journal on Multimodal User Interfaces, 1, 7-20.

[13] https://learn.microsoft.com/en-us/azure/ai-services/speech-service/speech-sdk

[14] https://learn.microsoft.com/en-us/azure/ai-services/speech-service/speech-to-text

[15] https://medium.com/@sujathamudadla1213/what-is-the-use-of-secure-filename-in-flask-9eef4c71503b

[14] McFee, B., et al. (2015). librosa: Audio and music signal analysis in python. Proceedings of the 14th python in science conference.

[15] https://librosa.org/doc/latest/index.html

[16] Xue, J., Zhou, H. Physiological-physical feature fusion for automatic voice spoofing detection. Front. Comput. Sci. 17, 172318 (2023). https://doi.org/10.1007/s11704-022-2121-6

[17] Idilio Drago, Flavio Miguel Varejão - Uma analise experimental de métricas de similaridade na classificação de séries temporais - Departamento de Informática, Centro Tecnológico – Universidade Federal do Espírito Santo (UFES)
Av. Fernando Ferrari s/n – CEP 29060-900 – Vitoria – ES – Brasil

[18] https://librosa.org/docplayground/0.7.2/generated/librosa.effects.preemphasis.html

[19] https://librosa.org/doc/main/generated/librosa.feature.mfcc.html

[20] https://www.researchgate.net/figure/The-spectrogram-top-and-chromagram-bottom-of-an-ascending-scale_fig3_314918556

Anexos/ Apêndices

- MANUAL DE UTILIZADOR APP ANTI-DEEPFAKES (Projeto 9)
- Código utilizado para implementação