The University of Jordan

School of Engineering
Department of Computer Engineering

AI Solution for Stress Management in the Work Environment

Supervisor:
Prof. Gheith Abandah

Author(s):

Tamara Maher El-Heet	0186363
Reem Wael Abu-Sbeitan	0195365

4 June 2023

Submitted in partial fulfillment of the requirements of B.Sc. Degree in Computer Engineering

This page is intentionally left blank

ETHICAL STATEMENT

We, the undersigned students, certify and confirm that the work submitted in this project report is entirely our own and has not been copied from any other source. Any material that has been used from other sources has been properly cited and acknowledged in the report.

We are fully aware that any copying or improper citation of references / sources used in this report will be considered plagiarism, which is a clear violation of the Code of Ethics of the University of Jordan.

We further certify and confirm that we had no external help without the approval of our supervisor and proper acknowledgment when it is due. We certify and affirm that we never at any point commissioned a 3rd. party to do the work or any part of it on our behalf regardless of the amount charged or lack thereof. We also acknowledge that if suspected and thereafter proven that we commissioned a 3rd. party to do any part of this project that we risk failing the entire project.

We certify and confirm that all results presented in this project are true with no manipulation of data or fraud, that any statistics done, or surveys collected are conducted with the highest degree of scientific fidelity and integrity, and that if proven otherwise, we risk failing the entire project. We acknowledge that for any data collected, we have taken all the steps necessary in applying for proper authorizations if deemed necessary, and that all user data collected is subject to the utmost degrees of privacy and anonymity.

Tamara El-Heet
June 4th, 2023

Reem Abu-Sbeitan
June 4th, 2023

This page is intentionally left blank

SUPERVISOR CERTIFICATION

I hereby certify that the students in this project have **successfully finished** their senior year project and by submitting this report they have fulfilled in partial the requirements of B.Sc. Degree in Computer Engineering. □

I hereby certify that the students in this project have not completed their senior year graduation project and **I do not approve** that they proceed to the discussion. □

I suspect that the students have **violated** one or more of the clauses in the **ethical statement** and I suggest that an investigation committee look into the matter. □

Prof. Gheith Abandah.

Signature:

This page is intentionally left blank

DEDICATION

We dedicate this project documentation to all those who have supported and inspired us throughout this journey. This project would not have been possible without the unwavering support and guidance of our professors, teachers, friends, and family. Their encouragement and belief in our abilities have been instrumental in our success.

We also want to express our deep appreciation and special thanks to our graduation project supervisor Prof. Gheith Abandah for his advice, support, and encouragement. Lastly, we are grateful to the Department of Computer Engineering and its members, for their support throughout the past five years.

SYMBOLS, ABBREVIATIONS, AND ACRONYMS

AI	Artificial Intelligence
CNN	Convolutional Neural Network
FT	Fourier Transform
FFT	Fast Fourier Transform
LR	Learning Rate
MIT	Massachusetts Institute of Technology
ML	Machine Learning
SDK	Software Development Kit
SER	Speech Emotion Recognition
WHO	World Health Organization

ABSTRACT

As technology became an important part of our lives, and communication with machines can be done nowadays by voice, speech emotion recognition has become of great interest in recent years. Speech emotion recognition has given machines the ability to recognize the emotion embedded inside an audio clip. Our aim was to use this ability to detect if an individual is approaching a state of stress.

In this project, we developed an AI solution for stress management in work environments. By analyzing voice characteristics derived from audio clips transformed into spectrograms, our model can accurately determine the user's current emotion and predicts if the user is about to get stressed or not. The result is later presented through a user-friendly application that was developed using Flutter. We were able to achieve an accuracy of 99% for the test set and 71% for the real-life data. One notable feature of our application is its versatility in accepting inputs from almost any language not only Arabic or English. Moreover, our application's applicability extends beyond work environments, for example, it can be used in educational institutions.

To conclude, our AI solution and accompanying application provide an effective means of stress management in work environments. The versatility in language acceptance and adaptability to different settings make it accessible and valuable to users worldwide. By leveraging voice characteristics analysis, our solution promotes self-awareness, enhances emotional well-being, and empowers individuals to proactively manage stress in various aspects of their lives.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

This page is intentionally left blank

CHAPTER 1
INTRODUCTION

Artificial intelligence pace of progress is incredibly fast. It can be used in almost all aspects of life to make them much easier. One of the fields in which AI can achieve a major qualitative leap is the human's health. Focusing on the mental health as it has always been an important part of our lives, AI can help to maintain a healthy mind due to its capabilities in understanding and analyzing the emotional, psychological, and social well-being if trained well.

According to the World Health Organization – WHO, the impact of COVID-19 on mental health cannot be underestimated. A great number of people reported to the organization that they are suffering from psychological distress and symptoms of depression, anxiety or post-traumatic stress because of the pandemic [1]. In other words, dealing with Covid-19 in the couple past years caused a person's mental health to become a wider concern because of stress and anxiety. People can recognize if a person is stressed or not, yet, in work environments everyone is certainly busy and cannot be there to check the workers' stress levels and overall feelings all the time. Therefore, we thought that AI would be a great solution for stress management in those cases.

Although mental health is a very sensitive topic, it is not getting enough attention in the Middle East. A person's stress level in work environments can affect the whole company's achievements, meaning that the mental health of a worker can either help in enhancing the company or cause it to fall apart. This is why we wanted to create an application that takes the worker's voice as input to detect his/her stress levels and notify him/her.

1.1. Problem Definition

Stress is a natural response that arises when individuals find themselves confronted with pressure or demanding situations. It is often triggered when a person experiences something new or unexpected that the person does not have good control over it. In work environments, it is the responses that occur when the requirements of the job do not match the capabilities, resources, or needs of the worker [2].

As university students, we have noticed that academic pressure is a constant presence in our lives. However, it was during our internship training this year that we truly experienced the weight of this pressure. This realization prompted us to explore more extensively the domain of stress and stress management specifically within work environments, aiming to gain a comprehensive understanding of these crucial aspects.

People deal with stress differently. Therefore, the ability to cope with stress and difficult situations varies among individuals and is influenced by various factors. These factors include early life experiences, personality traits, past experiences, and emotional well-being. Stress management has its very own field of research and solutions, such as meditation, deep breathing exercises, talking to someone, and many other ways to relax.

The main issue in these solutions is, there is no solution that alarms the person that he is under stress or that he is about to get stressed. The person will reach a high level of pressure which will develop unhealthy responses, and in some cases, it might cause mental or physical injuries. In many work environments, colleagues may not always be able to offer advice or support when a worker is experiencing stress. Often, they are occupied with their own responsibilities and may not realize the extent of someone's stress until it becomes overwhelming. This is where the role of AI and our application comes into play. We want our application to be able to provide early notifications to individuals, alerting them about their emotional states and preventing their stress from building up.

1.2. Proposed Solution

The proposed solution is an AI solution that analyze the voice characteristics to understand the individual's emotions. Since some emotions relate to stress directly, our AI solution will analyze the users' voice and inform them if they are about to get stressed or if they are in a safe state. This helps the users to manage their stress more effectively and take timely action.

Let us consider anger as an example: anger can cause or contribute to increased levels of stress. When individuals experience anger, it activates the body's stress response, triggering physiological changes such as increased heart rate, elevated blood pressure, and the release of stress hormones like cortisol. This physiological arousal associated with anger can intensify the overall stress levels in the body [3]. On the other hand, positive emotions such as happiness can have a positive effect on reducing stress levels. Happiness can promote relaxation, lower heart rate, decrease blood pressure, and reduce the release of stress hormones. It also enhances overall well-being and resilience, making individuals better equipped to cope with stressful situations. In order to have a better understanding of the proposed solution, Fig. 1 shows the main steps of our project.

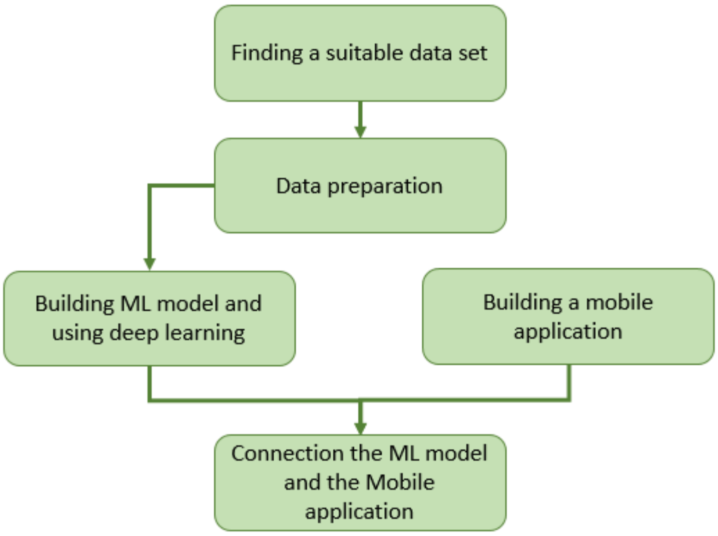

Finding a suitable data set
Data preparation
Building ML model and using deep learning
Building a mobile application
Connection the ML model and the Mobile application

Figure 1 – System Diagram

1.3. Project Deliverables

For the proposed AI Solution for Stress Management in the Work Environment, the following must be delivered:

1. A ML model that is built using deep learning techniques and is trained using the data set that we provided. The model must be able to know the emotions from the voice characteristics.
2. A mobile application that is built using Flutter. It is built using the Dart language and it is easy to use. The application will allow the user to upload an audio clip or to record an audio clip. The result of the emotion and if the user is stressed or not will then be shown to the screen.

1.4. Project Impact

1.4.1. Personal Impact

The personal impact of our project can be significant. Here are some specific personal impacts:

1. Enhanced Emotional Well-being: By actively monitoring stress, the project supports users in maintaining emotional balance.
2. Self-Awareness: The project can increase self-awareness since the users will know when they are about to get stressed. This awareness enables individuals to recognize their triggers, patterns, and the effects of stress on their well-being.
3. Empowerment: Users feel empowered to take control of their stress levels and overall well-being.
4. Improved Coping Strategies: When the users know when they are stressed, they can know what caused them to stress as well. Knowing what triggers them to get stressed will allow them to try and deal with the situation.

Looking at the personal impacts above, we can say that also the Mental and the physical health of a user can benefit a lot from our project since they will be affected positively.

1.4.2. Main impact

Which is the economic and social impact. In 2019, a study in the US showed that 70% of workers are stressed about their health, jobs, and finances. A further study showed that those 70% feels less productive and less engaged which indicated that worker stress costs employers billions due to lost productivity [4]. The same thing will definitely apply to other countries including Jordan, the more the workers stress the less productive they get.

The main impact of our project will be reducing the employees' stress, which will increase their productivity and motivate them to do better in their job. Furthermore, this will improve their mental health and make them feel happy and able to give. This will eventually reduce social and economic problems among individuals. If every employee in the organization works at its maximum capacity, this will reflect positively on the company's revenues.

1.5. Report Guide

Now that we have introduced the project concept, we will move on to discuss some related applications and papers in Chapter 2, this will help us to get a better understanding to enhance our project. After that, we will describe our application and we will figure out the right machine learning techniques to use, how to implement them, and benefit from them efficiently in Chapter 3. In Chapter 4 we will talk about the results of our testing process and discuss them. In conclusion, we will briefly review what we have discussed in our documentation, along with our future vision of how our project could be improved in Chapter 5.

CHAPTER 2
RELATED WORK

After figuring out the project idea, we had to do a couple of research on related and previous works on which the experimental work is based. Emotion recognition using Ai solutions is a wide field and many companies came up with great projects on it, here are some of them:

The first company we will be talking about is Affectiva, Affectiva is a company that specializes in emotion recognition technology. Their technology is designed to analyze facial expressions and detect emotions in real-time. The system uses machine learning algorithms to recognize patterns of facial movements and associate them with specific emotional states. Affectiva's emotion recognition technology can be used in a wide range of applications, including market research, advertising, and customer experience analysis. Affectiva's technology has also been used in healthcare to monitor patients' emotional states, as well as in education to help teachers to understand how their students are feeling. Additionally, the technology has been used in the gaming industry to create more immersive and responsive games that adapt to the player's emotions.

Affectiva also offers emotion recognition technology that analyzes voice characteristics which is closer to our project idea. Their technology uses machine learning algorithms to detect and classify emotions in speech, based on factors such as pitch, tone, and intensity. The voice-based emotion recognition technology has applications in a variety of industries, including customer service, healthcare, and market research. For example, it can be used to analyze customer interactions with call center representatives. However, their facial expression recognition technology is much better than their emotion recognition through voice characteristics. Fig. 2 shows one of Affectiva project which shows an example of their facial expression recognition technology.

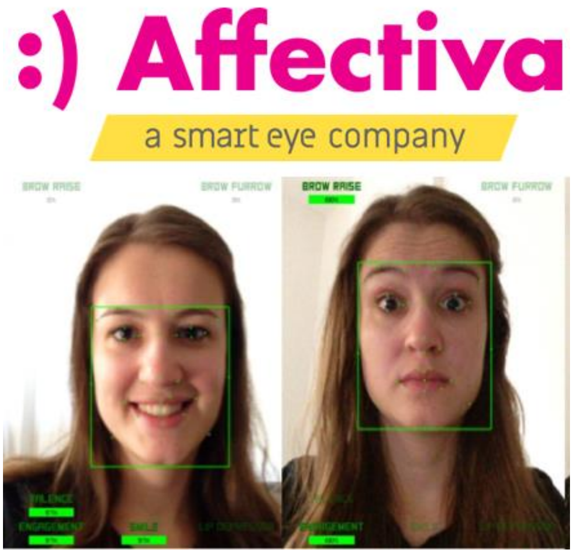


Figure 2 – Affectiva Inc. face recognition technology.

Another company is Emotient. Emotient was a company that developed emotion recognition technology, specifically focused on facial expression analysis. They used computer vision and machine learning algorithms to analyze facial expressions and detect emotions in real-time. Emotient's technology was primarily used in the advertising and market research industries, to help companies understand how consumers were reacting to their products and services. For example, it could be used to analyze how people responded to a particular ad campaign or product display in a store.

In January 2016, Apple acquired Emotient, and the technology was integrated into Apple's product line. However, in 2017, Apple shut down the Emotient division and stopped offering its technology to third-party clients. Overall, the technology developed by Emotient was similar to that of Affectiva. However, Affectiva has continued to develop and offer its technology to a wide range of clients, while Emotient's technology has been integrated into Apple's products and is no longer available to external clients. Emotient's primary focus was on facial expression analysis for emotion recognition, meaning that they did not offer voice-based emotion recognition technology.

The last company was Affective Computing Group at MIT. The Affective Computing Group at the Massachusetts Institute of Technology (MIT) is a research group that focuses on the development of intelligent systems that can recognize, interpret, and respond to human emotions. One of the main areas of research for the Affective Computing Group is the development of wearable sensors that can detect physiological signals related to emotion, such as changes in heart rate, skin conductance, and facial expressions. These sensors can be used to develop systems that can detect and respond to a user's emotional state in real-time.

Another area of research for the group is the development of affective computing applications in healthcare, education, and entertainment. For example, the group has developed systems that can help children with autism to recognize emotions, and systems that can monitor the emotional state of patients with mental health conditions. After reading about the previous companies and their amazing projects, we found out that their models and technologies are kept private. Therefore, we started looking at a stress-related voice recognition AI projects that are available or are currently being developed and researched.

Stress voice recognition involves analyzing vocal features such as pitch, tone, volume, and other vocal characteristics to detect stress in a person's voice. Here are some stress voice recognition AI projects:

1. Stanford University's Center for Interdisciplinary Brain Sciences Research - Researchers at this center are working on developing a machine learning model that can detect stress in a person's voice. They are using a dataset of voice recordings from individuals with and without anxiety disorders to train the model.
2. Sonde Health - Sonde Health is a company that is developing a voice-based diagnostic platform that can detect stress and other health conditions from a person's voice. They use machine learning algorithms to analyze vocal patterns and detect changes in vocal characteristics that are associated with stress.
3. Multimodal Technologies and Interaction Lab (MuTI Lab) - Researchers at this lab are working on developing a multimodal stress recognition system that can

detect stress from vocal features as well as physiological signals such as heart rate and skin conductance.

These are just a few examples of the stress voice recognition AI projects that are currently being developed. The field of stress voice recognition is still in its early stages, and new projects and technologies are being developed all the time. Unfortunately, there is no stress voice recognition application or project that is fully published to our day yet.

Furthermore, we tried to find a data set of voice recordings from individuals with and without anxiety disorders to train the model similar to the one that Stanford University's Center for Interdisciplinary Brain Sciences Research Center was using but we did not have access to that kind of dataset. As a solution, we found data that has different emotions, and we studied the relation between those emotions and stress in order to be able to predict if the individual is about to get stressed or not.

Lastly, we were able to find a project that is similar to our idea but for emotions only, such as Speech Emotion Recognition by Fine-Tuning Wav2Vec 2.0 [5]. The dataset that has been used in it provides 1440 samples of recordings from actors performing on 8 different emotions in English, which are: 'angry', 'calm', 'disgust', 'fearful', 'happy', 'neutral', 'sad', and 'surprised'. We studied this project to get an idea of where to start when building our project, which hyper parameters should we focus on when modifying the model, and an overall understanding on how to start dealing with audio based datasets.

CHAPTER 3
Background

In this chapter, we will present all the details about the solution and implementation to cover all the steps that we went through:

3.1. Solution Description

Our project as mentioned previously is an AI Solution for Stress Management in the Work Environment. The AI model that we created should be able to take an audio clip and predict the emotion of the user. Our project also consists of a Flutter application that allows the user to upload or record an audio clip, this clip will use the implemented AI model that we trained to predict the emotion and show it to the user while providing if the user is close to getting stressed or is in an emotionally safe state.

3.2. Dataset

Getting the dataset that we want to work with was one of the main challenges that faced us throughout this project. We first wanted to find an audio that consists of stressed and relaxed classes, unfortunately, the data related to this was private and we did not get the permission. Therefore, we changed the idea from detecting the stress levels of an individual to predicting if the individual is about to get stressed by knowing the individual emotion and its relation with stress.

The data that we chose was the MELD dataset which was available through GitHub. The MELD dataset is an enhanced and extended version of the EmotionLines dataset. It includes audio, visual, and text data from dialogues taken from the Friends TV series. The dataset consists of over 1400 dialogues and 13000 utterances, involving multiple speakers. Each utterance in the dialogue is labeled with one of seven emotions: Anger, Disgust, Sadness, Joy, Neutral, Surprise, and Fear. Additionally, sentiment annotations (positive, negative, and neutral) are provided for each utterance.

We chose this dataset because it is significant and reliable. The purpose of the MELD dataset in the first place was to provide a comprehensive and multimodal resource for studying emotion recognition and sentiment analysis in natural language processing and machine learning tasks. By incorporating audio, visual, and textual modalities, the dataset allows researchers to explore the combined influence of different modalities on emotion detection and sentiment analysis. Open-source datasets like MELD can also promote collaboration, transparency, and further advancements in research and development within the scientific community.

The MELD dataset can be valuable for a stress management project because it provides a diverse range of dialogues and utterances labeled with emotions and sentiment annotations. By utilizing this dataset, we can train and develop models that analyze and recognize stress-related emotions expressed in speech. The multimodal nature of the dataset, incorporating audio, visual, and textual information, allows for a comprehensive understanding of the user's emotional state. By incorporating the insights from the MELD dataset into our stress management application, we can enhance its accuracy and effectiveness in detecting and responding to stress-related emotions. That is the main reason why we chose to work with it. Fig. 3 shows an Example Dialogue from the dataset [5].

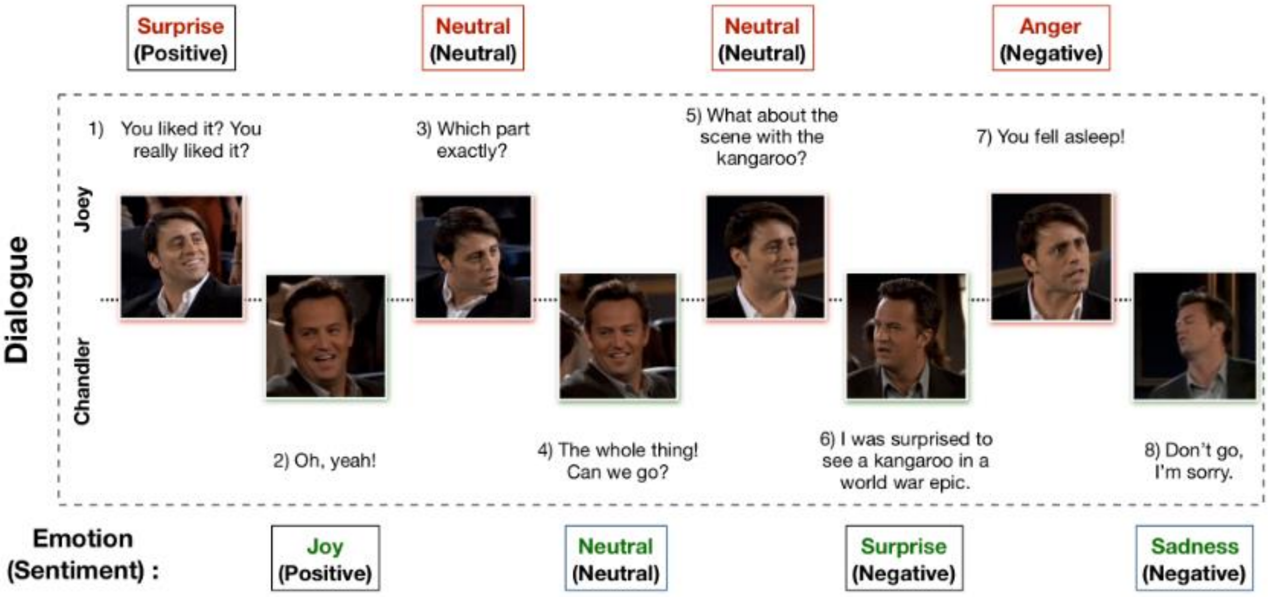
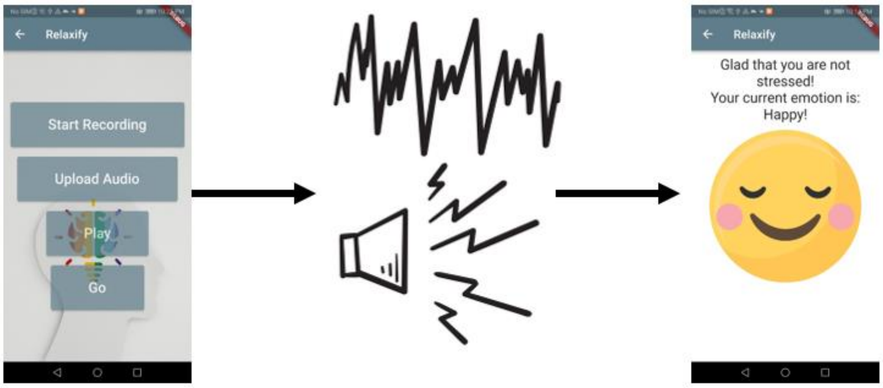


Figure 3 – Example Dialogue.

3.3. Design overview

Our proposed solution in this project is similar to a SER system in which a short audio clip will be analyzed, the feelings contained in this clip will be predicted and the emotion and its relation to stress will be returned as a result. To get the final result, the audio clip goes through several stages starting with the mobile application, where the clip can be uploaded or recorded, then it will be passed to the ML model which will analyze it based on the patters learnt from the collected dataset, determine the result, and return it to the application to show it to the end user. Fig. 4 shows a block diagram of the project.



Glad that you are not stressed!
Your current emotion is:
Happy!

The end user specifies the audio in the application.

The model analyzes the audio clip.

The result is shown in the application.

Figure 4 – Block Diagram of the System.

Initially, we began constructing the model using Jupyter Notebook. However, as our project progressed and we went deeper into it, we found that it is better to transfer our work to Google Colaboratory. This decision was primarily motivated by the convenience of easily mounting our project to Google Drive. By doing so, we ensured seamless access to our data and facilitated smoother collaboration among team members.

We installed the TensorFlow Hub library and imported essential modules such as OS, matplotlib, numpy, logging, and more. We needed these tools since they play a crucial role in our project by providing functions for data handling, visualization, numerical operations, and logging of important information. By including these necessary components, we ensured that our project has the required functionality and capabilities to proceed effectively.

Many programming languages can be used for machine learning, (e.g., R, Java, and Python). For our model, we chose to work with Python, which is considered today as one of the most popular programming languages in the world. Python has many advantages such as:

1. Open-source programming language: Python has a large community of developers. Also, it has a great documentation, which makes it easy to learn.
2. Powerful libraries and frameworks: Python have a collection of ready-to-use libraries and frameworks that provide a significant functionality and saves time and effort.
3. Simplicity: Python's simplicity and straightforward syntax allow people to understand it easily.

3.4. AI Model

Instead of creating a model from scratch, it was better to work with a pre-trained one, we utilized MobileNetV2, which is a widely recognized and powerful convolutional neural network architecture. MobileNetV2 is known for its efficiency and accuracy in various computer vision tasks. By leveraging MobileNetV2, we were able to benefit from its already-learned features, saving significant time and effort in the development process.

In other words, the goal of MobileNetV2 is to provide a highly efficient neural network model that can perform well on mobile and embedded devices with limited computational resources. MobileNetV2 is trained on large-scale image classification tasks, such as the ImageNet dataset, and has been shown to achieve competitive accuracy with much smaller model sizes compared to other popular architectures like VGG or ResNet. It strikes a good balance between model size, computational efficiency, and performance, making it suitable for a wide range of computer vision applications including our project.

Our project deals and works with audio data, yet the MobileNetV2 is a convolutional neural network (CNN) architecture designed for image classification. We chose to use MobileNetV2 to provide our application with a unique advantage. Instead of processing language-specific information, MobileNetV2 allows us to analyze voice characteristics extracted from spectrograms—an image representation of audio—and detect emotions without relying on language comprehension. This approach enables our application to accurately assess emotions in a language-agnostic manner, enhancing its versatility and applicability across diverse audio datasets.

Spectrograms had a huge role in our project, so we will now explain what we learned about it. A spectrogram is a visual representation of the frequency content of a signal over time.

It provides a detailed view of how the frequencies in an audio signal change over different time intervals. Spectrograms are commonly used in audio processing and analysis tasks, including speech recognition, music analysis, and sound classification.

The spectrogram is created by performing a Fourier Transform on short segments of the audio signal, typically using the Fast Fourier Transform (FFT) algorithm. The result is a set of frequency components or bins that represent the intensity or magnitude of each frequency present in the signal at that particular time frame. The spectrogram displays these frequency components as a 2D plot, where the x-axis represents time and the y-axis represents frequency. The intensity of each frequency component is often depicted using colors or grayscale, with brighter colors indicating higher amplitudes or stronger presence of that frequency. Fig. 5 shows a Fast Fourier Transform signals representation.

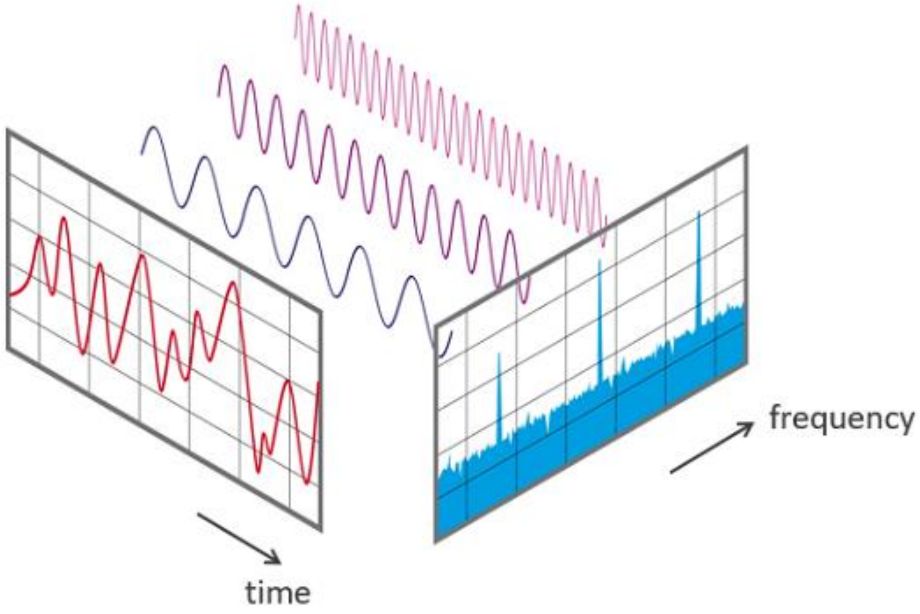


Figure 5 –Fast Fourier Transform

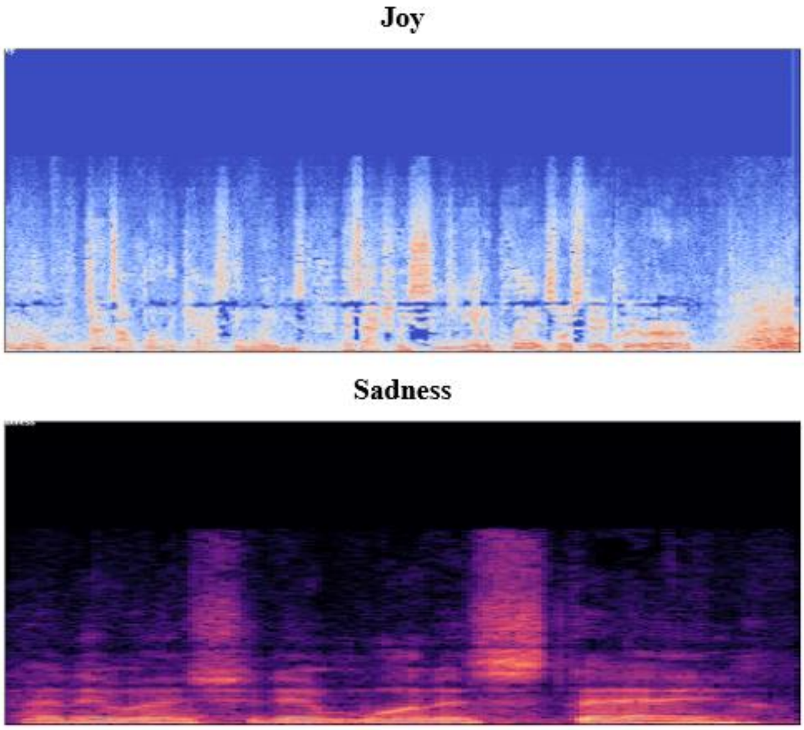
Spectrograms are particularly useful in machine learning applications as they provide a compact and informative representation of audio signals. They can be fed into deep learning models, such as convolutional neural networks (CNNs), for tasks like speech recognition, emotion detection, or audio classification.

Furthermore, spectrograms play a crucial role in emotion detection from audio signals. Emotions are often conveyed through changes in vocal characteristics such as pitch, intensity, and timbre. By analyzing the frequency content of an audio signal using spectrograms, it becomes possible to extract relevant features that reflect these vocal characteristics and correlate them with different emotions. Spectrograms are typically used to convert the audio signals into visual representations that highlight the frequency patterns over time. These spectrograms capture important information about the variations in the vocal characteristics associated with different emotional states.

Machine learning algorithms, such as convolutional neural networks (CNNs), can be trained on spectrogram data to learn the relationships between the frequency patterns and corresponding emotions. The CNNs can automatically extract relevant features from the spectrograms and make predictions about the emotional states expressed in the audio. By utilizing spectrograms in emotion detection, the application can analyze vocal cues and identify emotional states in an audio recording, regardless of the language used. This language-agnostic approach enables the detection of emotions across different cultures and languages, making it a powerful tool for cross-cultural and multilingual emotion analysis.

Overall, spectrograms are a valuable tool in audio analysis, providing a visual representation of the frequency content of an audio signal over time. They enable the extraction of meaningful features for various applications in speech, music, and sound processing.

We trained our model using a dataset consisting of 6,597 audio clips, each with a duration of 3 seconds. To convert these audio clips into spectrograms, we developed a Python code with the assistance of ChatGPT. As a result, we obtained a collection of 6,597 spectrogram images, which were used for training our model. The data had seven emotions: Surprise, joy, anger, neutral, sadness, fear, and disgust. Fig. 6 shows the spectrograms of sadness and joy emotions, While Fig. 7 shows the spectrograms of surprise and anger.



Joy

Sadness

Figure 6 – Joy and Sadness spectrograms.

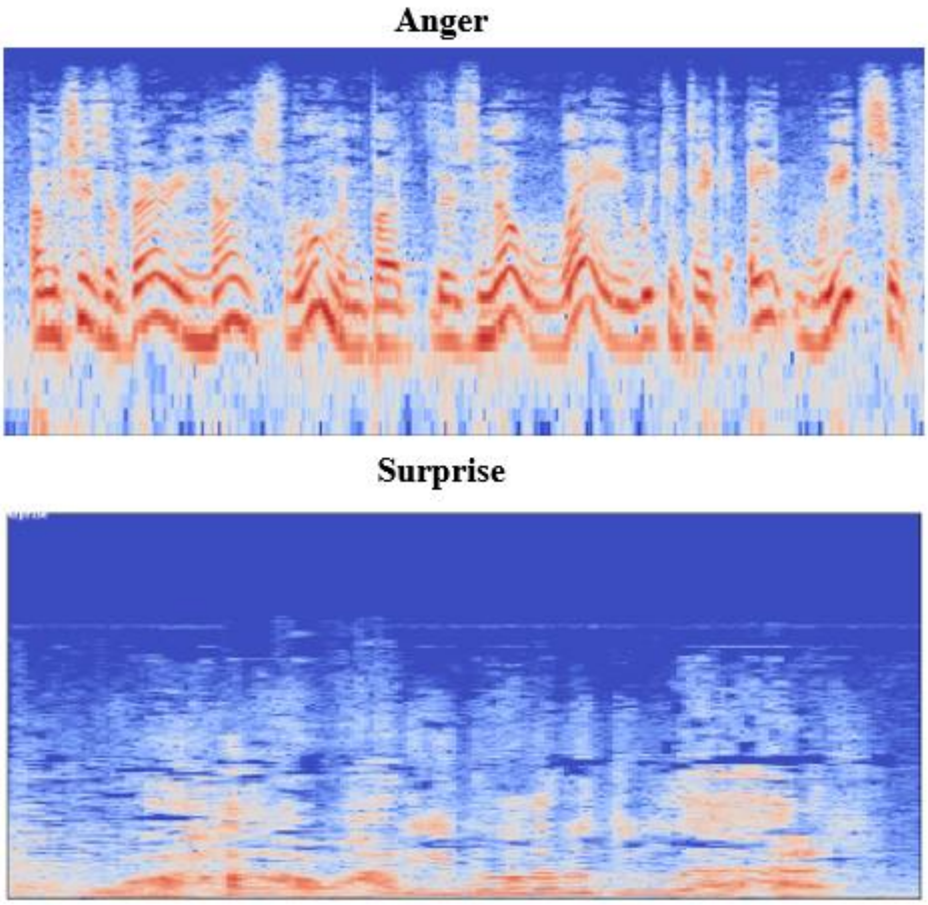


Figure 7 – Anger and surprise spectrograms.

We have generated spectrograms for a variety of emotions, enabling us to explore the distinct acoustic characteristics associated with each emotional state. Let us focus on the spectrogram of anger and delve into its notable features. The spectrogram of anger reveals prominent patterns that differentiate it from other emotions. Upon visual analysis, several key observations can be made. Anger spectrograms often exhibit intensified energy concentrated in lower frequency ranges. This heightened intensity can manifest as elevated amplitudes and increased power within specific low-frequency bands, indicating a strong and forceful vocal expression.

Furthermore, anger spectrograms may demonstrate consistent and sustained energy throughout the duration of the audio signal. This sustained energy is characterized by extended horizontal bands of elevated intensity, signifying the persistence of anger-related vocalization.

In addition to the lower frequency emphasis, anger spectrograms might display sharper and more pronounced transitions between frequency components. These rapid shifts in intensity and frequency content can reflect the aggressive and explosive nature of the emotional state. Fig. 8 shows the spectrogram of Anger with frequencies and color intensities.

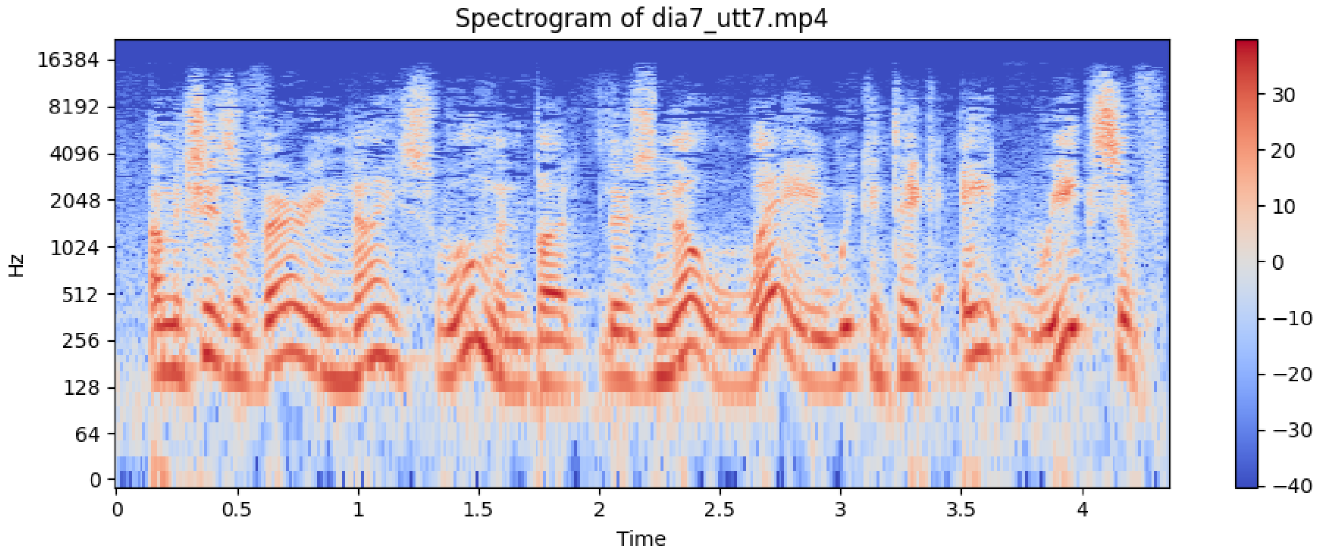


Figure 8 – Anger spectrogram.

3.4.1. Convolutional Neural Networks (CNN):

Convolutional Neural Networks play a crucial role in image classification tasks, including the recognition of emotions such as surprise, joy, anger, neutral, and sadness which are the final emotions in our dataset. CNNs excel at learning the weights and biases associated with various objects within input images, enabling accurate differentiation between these emotions. CNN architectures typically consist of multiple layers, with the primary building block being the convolutional layer. Following the convolutional layers, pooling layers are often introduced to downsample the features extracted by the convolutions. These convolutional and pooling layers work together to capture high-level features, such as edges, textures, and patterns specific to each emotion.

In our CNN model, the input layer accepts a 2D colored image, typically of size 224 by 224 pixels. The output layer predicts the final emotion classification, which can be one of the five emotions: surprise, joy, anger, neutral, or sadness. Between the input and output layers, hidden layers perform complex computations, allowing the network to learn and extract meaningful representations of the input data.

Each pixel of the input image serves as an input to individual neurons in the input layer. Neurons within the network are interconnected, forming multiple hidden layers. These connections are assigned weights, which determine the strength of influence from one neuron to another. Additionally, biases associated with each neuron are added as linear combinations to the weighted inputs.

After these computations, the resulting activations from the hidden layers contribute to the subsequent layers without explicitly applying an activation function. This allows the network to model complex relationships between the input data and the corresponding emotions without introducing further non-linearity. Each neuron's activation still determines its level of contribution to the subsequent layer based on the weighted inputs and biases.

During forward propagation, data flows through the network, and predictions are made based on the highest activation values in the output layer. The output layer provides probabilities for each of the five emotions, enabling a probabilistic understanding of the predicted emotion.

3.4.2. Transfer Learning

For our model, we employed Transfer Learning, which involves utilizing a pre-trained model to address a different but related problem. This approach proves beneficial when training neural networks with limited datasets, significantly reducing the training time.

During Transfer Learning, the input layer and hidden layers of the pre-trained model remain unaltered. However, the output layer needs to be customized to match the classes of the new problem. To accomplish this, we selected a pre-trained model based on MobileNet neural networks within the Tensorflow framework as mentioned previously.

Two additional steps were undertaken to adapt the pre-trained model to our specific task. Firstly, we adjusted the output layer to accommodate five classes. Then, we performed fine-tuning, which involves retraining the model's parameters to better align with the new task. By setting the "trainable" parameter to true before fitting the model, we enabled the fine-tuning process. When training the model, we specified the number of epochs, which signifies one complete cycle of training the neural network with the entire training dataset. The suitable number of epochs depends on the input data size. An inappropriate number of epochs may result in over-fitting, where the model captures unnecessary features or noise, or under-fitting, where the model fails to learn the data adequately. In our case, we used seven epochs. Once the model was trained, we saved it in .h5 format.

Alternative approaches exist for speech emotion recognition models, such as audio classification. This method involves extracting features from the audio file, generating arrays of numbers that represent various sound signal features, including MFCC, chroma, rms, zero-crossing rate, and others. This feature extraction process can be performed using specialized tools like MATLAB or Python libraries like Librosa, which offer convenient functions for feature extraction. The extracted feature arrays serve as inputs to the model. However, this approach typically requires a substantial amount of data to achieve acceptable accuracy.

We experimented with this audio classification approach on our dataset and also explored transfer learning by extracting features and training a shallow model on them. Ultimately, we concluded that for small datasets, using the audio file as a spectrogram input directly to the model outperformed passing the extracted features using Librosa functions.

3.5. Functional and Non-Functional Requirements

Functional Requirements:

The functional requirements of our project can be summarized as the following:

1. Emotion Recognition: The application can predict the emotion in a given audio file. By assessing the user's emotional state, it can determine whether the user is approaching a state of stress or not.

2. Upload: The application gives the users the ability to upload audio file from phone local storage.
3. • Record: The application gives the users the ability to record an audio clip.

Non-Functional Requirements:

1. Ease of use: The application is user friendly and has a very simple, colorful, and easy to use interface that will help users to use it easily.
2. Performance: The application must give users the requested results in a reasonable amount of time.
3. Reliability: The application must give users the ability to use its functionality in multiple scenarios without errors or crashing.
4. Accessibility: The application is accessible through smart devices with an android operating system.
5. Multilingual Interface: The application can take any language as an input, not only Arabic and English.

3.6. Flutter Application Implementation

In this section, we will provide a straightforward analysis of our proposed application and talk about how we created the user interface. First, we will explain the tools and software we use to develop the application. Then we will discuss different diagrams, like the use case diagram and architecture diagram. Finally, we will show you the design of the application and its user interface.

3.6.1. Android Studio

Our application was developed using Android Studio. We chose to develop our application using Android Studio for several reasons. Firstly, Android Studio is the official Integrated Development Environment (IDE) for Android app development. It provides a comprehensive set of tools and resources specifically designed to streamline the process of creating Android applications.

One of the key advantages of Android Studio is its robust and intuitive user interface. The IDE offers a user-friendly environment with a wide range of features and functionalities that make it easier to design, code, and test our application. It provides a visual layout editor, code editor with intelligent suggestions, and a powerful debugging tool, all of which contribute to a smooth and efficient development experience.

Moreover, Android Studio offers seamless integration with other essential tools and services. It supports the Android Software Development Kit (SDK) and allows easy access to a vast library of pre-built components, frameworks, and APIs. This integration enables us to leverage a wealth of resources to enhance our application's functionality and performance.

Overall, we could have used Visual Studio Code or other IDEs, but Android Studio emerged as the ideal choice for our application development due to its status as the official Android development IDE, its user-friendly interface, and its seamless integration with essential tools and services.

3.6.2. Flutter

We used the Flutter SDK (Software Development Kit) in our application development. Flutter is a powerful framework that supports building applications for multiple platforms. It provides a lot of useful features, such as ready-to-use user interface components that can be customized to create attractive and interactive designs.

For our Android application, Flutter was a great choice because it has excellent support for audio input. It has a wide range of libraries and plugins specifically designed for audio processing and input. This made it easy for us to include features like audio recording, playback, and uploading. Flutter also simplifies the process of designing visually appealing interfaces that are compatible with various screen sizes, including mobile and web platforms. This capability allowed us to create consistent and attractive user interfaces from a single codebase.

To summarize, Flutter offers numerous benefits to app developers. Its ability to design beautiful interfaces that adapt to different screen sizes, along with the versatility of the Dart language, provides developers with a great development experience. Flutter developers can create visually appealing, cross-platform applications for both Android and iOS, enhancing efficiency and reducing development efforts.

To demonstrate the application's functionality and its interactions with other entities, we utilized a Use Case Diagram. In Figure 9, we present the Use Case Diagram that illustrates the various use cases of our application.

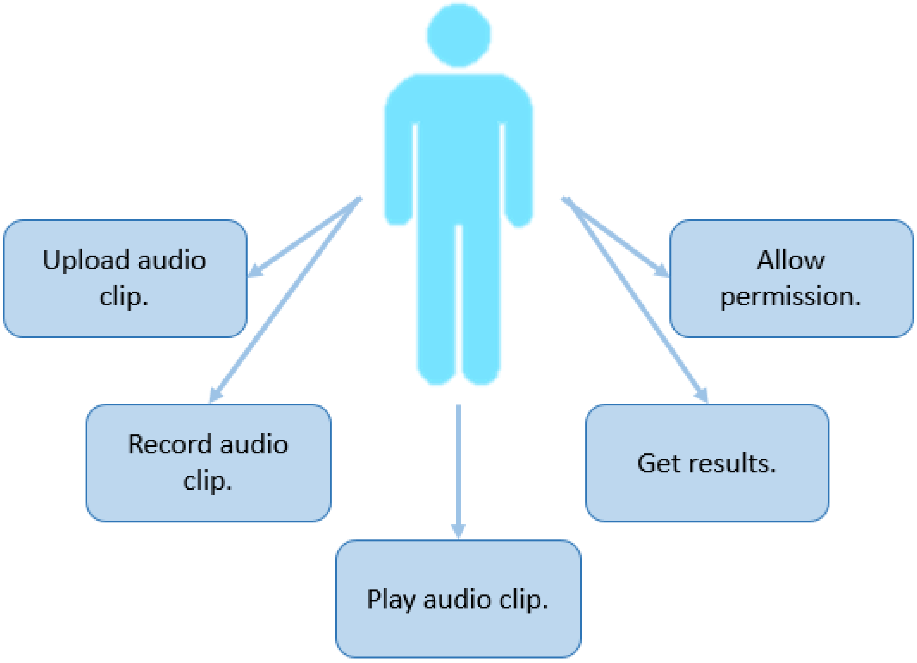


Figure 9 – Use Case Diagram.

Mobile application architecture refers to a set of techniques, processes, and patterns to develop a mobile application that meets both the business requirements as well as main functionalities of the application. Fig. 10 illustrates the Architecture Diagram of our application:

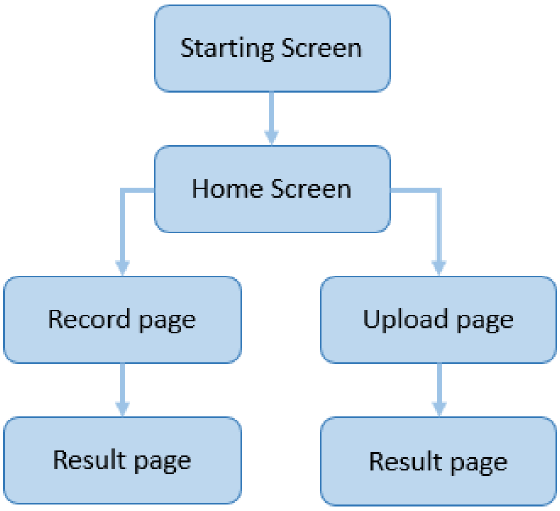


Figure 10 – Architecture Diagram.

Our second step in building this application was to develop the user interface of this application so that it performs its role in the ease of users' interaction with this application and its features, content, and functions. In this section we will show the pages of this application that are designed to achieve the project goals for the intended users. We have chosen the name "Relaxify" for our application, which stems from the term "Relax" and aligns with our primary goal. Our application aims to empower users by helping them recognize their emotions, particularly when they are on the verge of experiencing stress, in order for them to take proactive measures to prevent it. The name "Relaxify" encapsulates the essence of achieving relaxation and tranquility through our app's features and functionalities.

In addition to the name, we have carefully designed the application's logo, as depicted in Figure 11. The logo incorporates a small circle positioned on the face, with a deliberate focus on the relaxed eye. This design choice symbolizes our application's core focus on understanding and addressing if the individual is about to get stressed. By highlighting the relaxed eye, we visually emphasize our commitment to supporting users in managing their stress effectively.

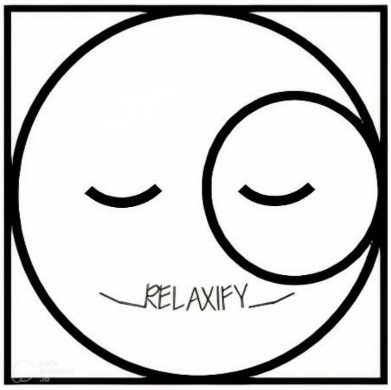


Figure 11 – The application's logo.

Next, we will introduce the application's startup page, which features a simple and minimalist design. The page primarily consists of a single button strategically positioned on a background displaying an image of a relaxed individual, as depicted in Figure 12. This serene visual perfectly captures the essence of the app, setting the tone for a calming and stress-relieving experience.

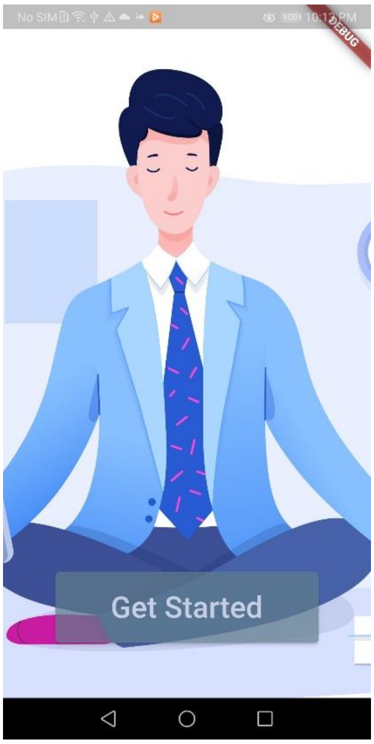


Figure 12 – Get started page.

Following that, upon pressing the "Get Started" button, the user will be directed to the home page. The home page of our application features four distinct buttons, each serving a specific purpose. These buttons provide convenient options for the user to perform the required actions.

1. The first button enables the user to record an audio clip directly within the application. By tapping on this button, users can capture their voice.
2. The second button facilitates the uploading of audio clips from external sources. This option allows users to select and import audio files stored on their device for further processing and analysis.

3. The third button grants users the ability to play back the recorded or uploaded audio clips. This feature ensures that users can listen to their audio recordings and review the content as desired.
4. Lastly, the fourth button offers a seamless transition to the result page. By selecting this button, users can access the page where they will find the analyzed results and relevant information based on the recorded or uploaded audio clip.

In summary, the home page of our application presents four distinct buttons that enable users to record audio, upload clips, play recorded or uploaded clips, and navigate to the result page for further insights. As shown in Figure 13, these options provide a user-friendly interface, enhancing the overall user experience and facilitating the efficient use of the application's key features.

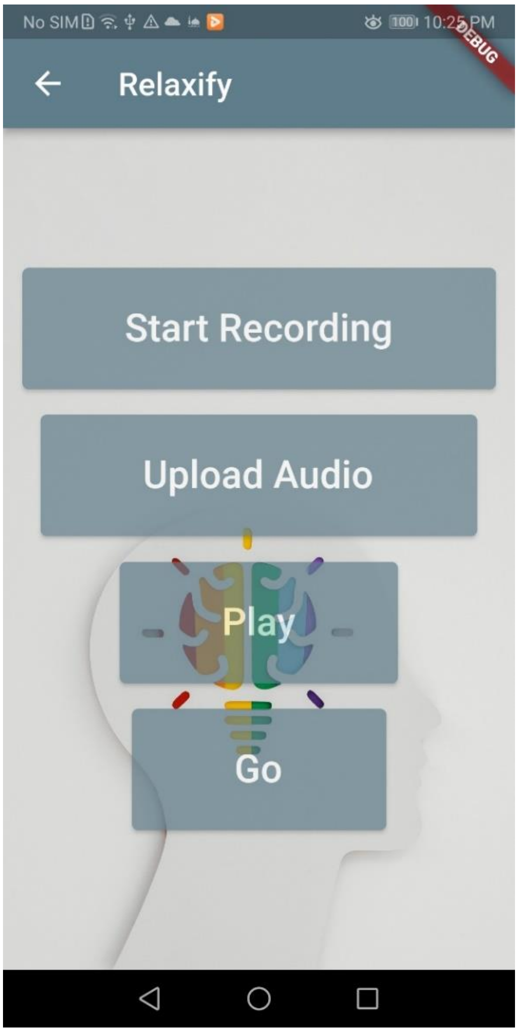
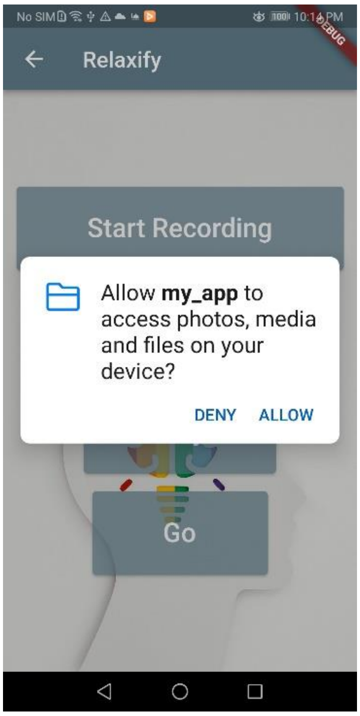


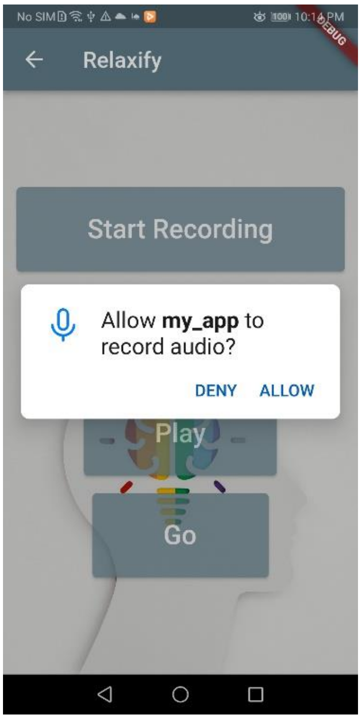
Figure 13 – Home page.

Subsequently, the application will prompt the user to grant necessary permissions to access files and media on their device, as well as permission to record audio. To ensure a seamless user experience, the application seeks these permissions to enable the smooth functioning of its features. Figure 14 illustrates the permission request to access media, while Figure 15 showcases the permission request for audio recording.



Allow **my_app** to access photos, media and files on your device?

Figure 14 – Photos, media, and files permission request.

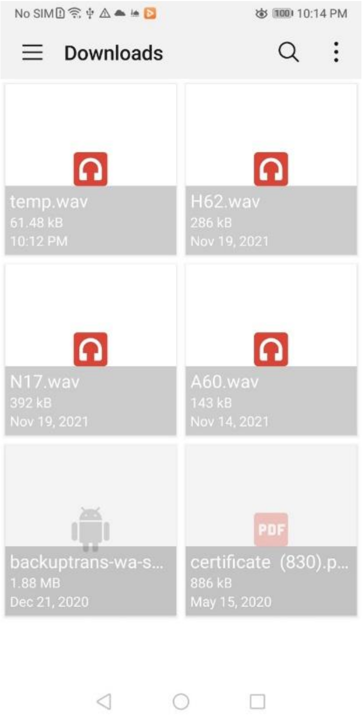


Allow **my_app** to record audio?

Figure 15 – Permission to record audio.

When the user clicks on the "Upload Audio" button, they will gain the ability to select and upload any audio file from their device. The application allows users to choose audio files in the .wav format for compatibility and seamless integration with the analysis and processing algorithms. Figure 16 shows the page of uploading an audio file.

By enabling users to upload audio files, the application offers flexibility in the types of content that can be analyzed and utilized within the app. Users can access their personal collection of .wav audio files, including recorded voice memos, music tracks, or other sound recordings, to be processed and utilized within the application's features.



temp.wav
61.48 kB
10:12 PM

H62.wav
286 kB
Nov 19, 2021

N17.wav
392 kB
Nov 19, 2021

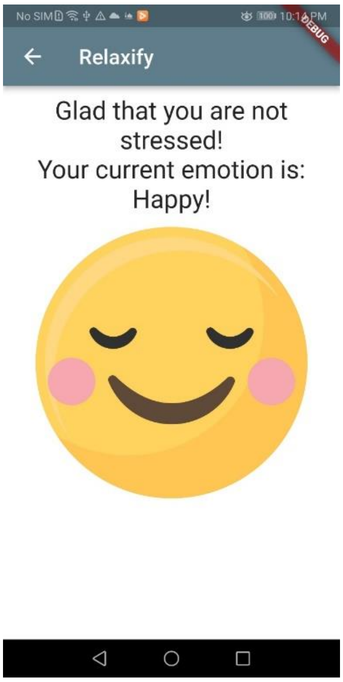
A60.wav
143 kB
Nov 14, 2021

backuptrans-wa-s...
1.88 MB
Dec 21, 2020

certificate (830).p...
886 kB
May 15, 2020

Figure 16 – Uploading audio files from device.

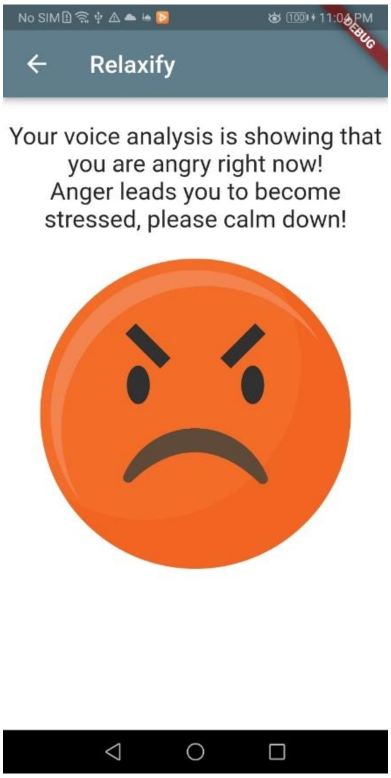
Figure 17 represents an example screen that users will encounter if their voice analysis indicates a state of happiness. The design of this screen visually conveys that the user is not currently experiencing stress, as happiness and stress are depicted as being far apart. This representation aims to provide users with a clear understanding that their current emotional state is positive and free from stress. The screen serves as a reassuring message to the user, indicating that they are in a relaxed and content state. It reinforces the idea that their emotional well-being is in a healthy condition based on the analysis of their voice. By presenting this information visually, users can easily interpret and interpret the results, promoting greater awareness and understanding of their emotional state.



Glad that you are not stressed!
Your current emotion is:
Happy!

Figure 17 – Current emotion: Happy!

Figure 18 represents an example screen that users will encounter if their voice analysis indicates a state of anger. The screen design visually signifies a potential indication of stress, as anger is closely related to stress. This depiction suggests that the user may be experiencing some level of stress due to their current emotional state.

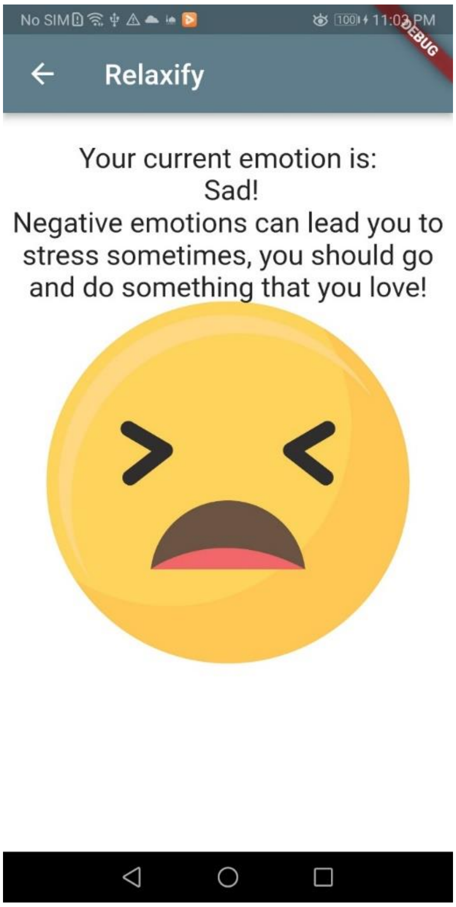


Your voice analysis is showing that you are angry right now!
Anger leads you to become stressed, please calm down!

Figure 18 – Current emotion: Angry!

Figure 19 represents an example screen that users will encounter if their voice analysis indicates a state of sadness. The design of this screen visually conveys the user's emotional state, indicating that they may be feeling sad and potentially experiencing stress. The depiction signifies the close relationship between sadness and stress, suggesting that the user might be in need of support and intervention.

The screen serves as a compassionate message to the user, acknowledging their emotional state and offering empathy. It encourages users to take steps toward addressing their sadness and managing any associated stress since they will now be aware of their current emotional state.



Your current emotion is:
Sad!
Negative emotions can lead you to stress sometimes, you should go and do something that you love!

Figure 19 – Current emotion: Sad!

CHAPTER 4
RESULTS AND DISCUSSION

By the end of this project, we successfully developed a mobile application that fulfilled all the intended functional and non-functional requirements We trained it a lot using different audio clips, mostly in English but also some in Arabic. It did really well, with a 99% accuracy on the test and 71% on new data. In the following sections, we will show the test results and the process that we went through.

4.1. Model testing and accuracy

To prepare the data for training, we converted our audio dataset into spectrograms. Spectrograms serve as visual representations that capture the frequency content of audio signals across time, enabling our model to extract features and patterns from the audio data. This approach will allow us to use the image processing model –which is MobileNet_V2- effectively to satisfy our goal.

As previously mentioned, one of the primary advantages of using spectrograms is that they are language-agnostic. Spectrograms rely solely on voice characteristics and do not require an understanding of the specific language used by the user. By focusing on voice attributes such as pitch, tone, intensity, and rhythm, spectrograms provide a language-independent representation of the audio data. This characteristic allows our application to accurately analyze emotions regardless of the user's spoken language.

During the initial training phase of our model using the spectrogram dataset, we encountered underfitting. The underfitting led to weak performance with an accuracy of only 36%. Underfitting occurs when a model fails to capture the complexity and patterns present in the underlying data. In this case, our model was unable to sufficiently learn from the spectrogram representations, resulting in inadequate predictive capabilities.

To tackle the underfitting issue, we increased the number of samples in our dataset. Which resulted in improving the model's accuracy, making it 60%. By expanding the dataset with additional examples, we provided the model with a larger and more diverse set of training instances to learn from. Continuing the development of our model, we focused on fine-tuning the hyperparameters to further enhance its performance. Parameters such as the learning rate, batch size, and network architecture were carefully adjusted and optimized to improve the model's accuracy. Through this iterative process, we were able to achieve an accuracy of 80%.

By finding an optimal balance between the learning rate, which determines the step size during training, the batch size, which affects the number of samples processed at each iteration, and the network architecture, which defines the structure and complexity of the model, we were able to maximize the model's predictive capabilities. The fine-tuning process involved a lot of experimentation and evaluating the model's performance under various parameter configurations. By adjusting and optimizing the hyperparameters, we found the combination that improved the model's accuracy up to 99%.

We mentioned before that we used the MELD dataset, we had to check the emotions distribution in it which is shown in Figure 20.

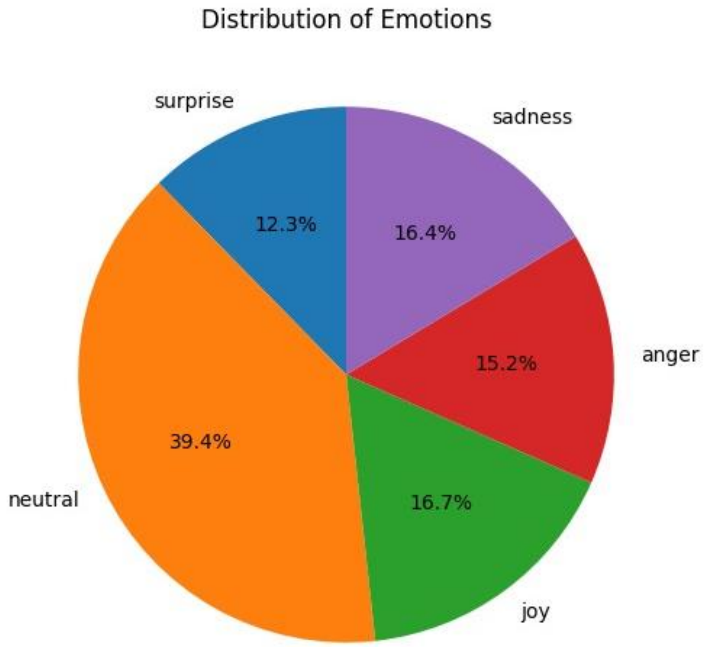


Figure 20 – MELD dataset emotion distribution.

The final results of the test set accuracy are shown in Table 1.

Table 1– Test set accuracy with lr=0.002.

	precision	**recall**	**f1-score**	**support**
surprise	1.00	1.00	1.00	145
joy	1.00	1.00	1.00	205
anger	0.97	0.94	0.96	189
neutral	1.00	1.00	1.00	478
sadness	0.95	0.97	0.96	195
accuracy			0.99	1212
macro avg.	0.98	0.98	0.98	1212
weighted avg.	0.99	0.99	0.99	1212

We will now be mentioning real-life data which we considered the external evaluation data. It is basically a mix of 30% MELD dataset, and 60% of data that we got from a dataset called RAVDESS [6] and an Arabic dataset which is called ANAD [7]. The emotion distribution of the external evaluation data is shown in Figure 21.

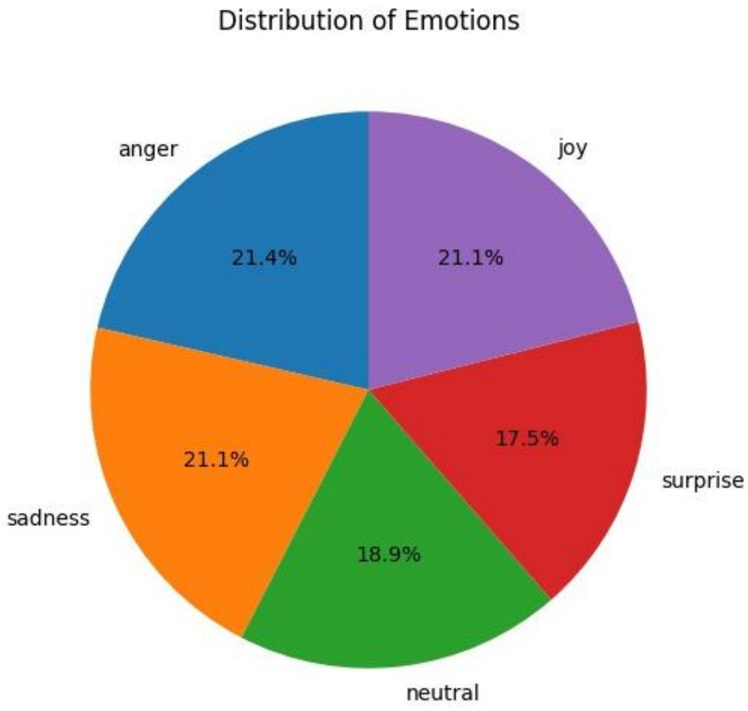


Figure 21 – External data emotion distribution.

During the real-life testing of our model, we observed a significant decrease in accuracy, resulting in a mere 40% success rate. This disparity can be attributed to two primary factors: the inadequate representation of specific emotions in the training data and the presence of dataset shift.

The first factor, insufficient representation of certain emotions in the training data, means that the model has not been exposed to a diverse enough range of samples for those particular emotions. As a consequence, when faced with real-life scenarios involving these emotions, the model struggles to accurately classify and predict them. While the second factor, dataset shift, occurs when there are differences in the statistical properties or characteristics between the training data and the real-world data encountered during testing or deployment. Dataset shift can arise due to various factors; such as changes in the data collection process or variations in the environment. These differences can impact the model's performance, as it may struggle to generalize from the training data to real-life scenarios, leading to a significant drop in accuracy.

To tackle the problem of insufficient data for certain emotions, we focused on increasing the representation of those emotions within the dataset. By augmenting the data specific to those emotions, we achieved a high accuracy of 98% on the test set. However, the accuracy was 65% in real-life testing, highlighting the challenge of overcoming dataset shift. Before getting 65%, the accuracy was 58% as shown in Table 2 down below.

Table 2– real-life data accuracy with lr=0.02.

	precision	**recall**	**f1-score**	**support**
surprise	1.00	0.85	0.92	65
bad	0.23	0.97	0.37	37
joy	1.00	0.82	0.90	60
anger	0.54	0.23	0.32	61
neutral	1.00	0.80	0.89	54
sadness	0.39	0.12	0.18	78
accuracy			0.58	355

We kept changing the hyperparameters until we finally achieved an accuracy of 65% with a learning rate equal to 0.002 and a number of epochs equal to 7. The results are shown in Table 3.

Table 3– real-life data accuracy with lr=0.002.

	precision	**recall**	**f1-score**	**support**
surprise	1.00	0.85	0.92	65
bad	0.00	0.00	0.00	37
joy	1.00	0.87	0.93	60
anger	0.39	0.46	0.42	61
neutral	1.00	0.81	0.90	54
sadness	0.39	0.67	0.49	78
accuracy			0.65	355

The maximum accuracy that we got from experimenting with the hyperparameters was slightly higher. We got an accuracy of 66% as shown in Table 4.

Table 4– real-life data accuracy with lr=0.001.

	precision	recall	f1-score	support
surprise	1.00	0.82	0.90	65
bad	0.00	0.00	0.00	37
joy	1.00	0.83	0.91	60
anger	0.76	0.21	0.33	61
neutral	0.96	0.81	0.88	54
sadness	0.41	0.97	0.58	78
accuracy			0.66	355

In an attempt to mitigate the impact of the dataset shift, we decided to merge the "fear" and "disgust" emotions into a single class named "bad." This approach aimed to reduce the number of distinct emotional categories and potentially improve the model's ability to generalize. However, the maximum accuracy we achieved with this approach was 67%, which was only slightly better than the previous real-life testing results.

After assessing the effectiveness of the merged class, we determined that it did not contribute significantly to enhancing the accuracy of the model. Therefore, we made the decision to eliminate the "bad" class and instead focus on the original emotional categories. By doing so, we observed a notable improvement in the model's performance, with an accuracy of 71%. This outcome demonstrates the positive impact of refining the emotional categories and aligning them with the model's training objectives, resulting in more accurate predictions. The accuracy results of the final model are presented in Table 5.

Table 5– real-life data accuracy with SGD optimizer, and lr=0.002.

	precision	recall	f1-score	support
surprise	1.00	0.88	0.94	50
joy	1.00	0.87	0.93	60
anger	0.47	0.13	0.21	61
neutral	1.00	0.81	0.90	54
sadness	0.42	0.90	0.57	60
accuracy			0.71	285
macro avg.	0.78	0.72	0.71	285
weighted avg.	0.76	0.71	0.69	285

We will now sum up all the real-life data models that we presented so far in Table 6.

Table 6– real-life data results.

Model	Learning rate	Number of epochs	Batch size	Accuracy
Model 1	0.02	10	64	58%
Model 2	0.002	7	64	65%
Model3	0.001	7	64	66%
Model 4	0.002	7	64	67%
Final model	0.002	7	64	71%

Including the confusion matrix in our model was essential for assessing its overall performance. The confusion matrix provides a detailed breakdown of the model's ability to correctly classify instances for each class and identify any instances of misclassification.

The confusion matrix can be used to pinpoint specific classes where the model may struggle, leading to misclassifications or lower performance. For this project, we generated two confusion matrices to assess the performance of the model. The first one was created on the test set and is shown in Figure 22.

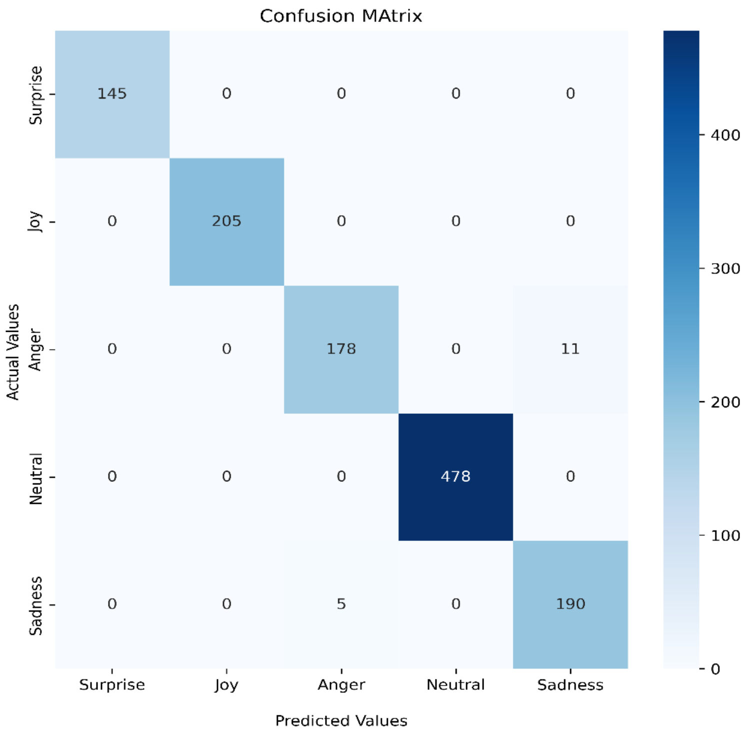


Figure 22 – Test set Confusion Matrix.

The test set matrix demonstrated a highly desirable outcome, as the diagonal of the matrix contained the majority of instances, indicating accurate classification by the model. This near-perfect diagonal alignment suggests that the model performed exceptionally well on the test set and successfully identified instances belonging to their respective classes with high precision.

The second matrix was based on real-life data that the model had not encountered before. From this matrix, we observed that the model faced challenges in accurately classifying instances belonging to the anger class. The diagonal value corresponding to anger was notably low, indicating a significant number of misclassifications. Additionally, we noticed that many instances originally belonging to anger were instead classified as sadness, which suggests a tendency for the model to confuse these two emotions. We can notice that from Figure 23.

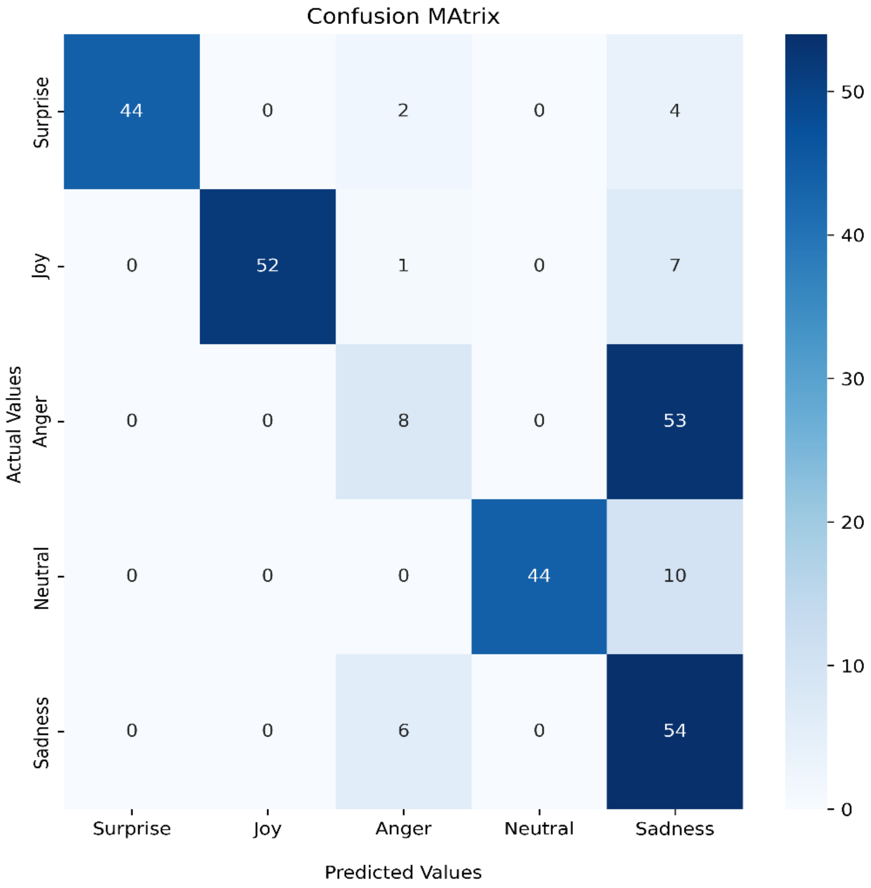


Figure 23 – Real-life data Confusion Matrix.

4.2. Other languages.

We used spectrograms as an input for MobileNet_V2 model instead of using SER model in order to allow our AI solution to work regardless of the user's language. So we decided to test our model which has only seen Arabic and English data on different languages. We used samples from a data set called ASVP-ESD [8].

ASVP-ESD is a shortcut for Speech and Non-Speech Emotional Sound. This dataset has 13285 audio files collected from movies, TV shows, and YouTube. It is 2 GB and has 12 different natural emotions (boredom, neutral, happiness, sadness, anger, fear, surprise, disgust, excitement, pleasure, pain, and disappointment) with 2 levels of intensity. It includes many languages such as Chinese, English, French, Russian and others.

When we first tried to test a Chinese sample, we got almost all the results wrong! We thought that it was because of the length of the audio clip, since our model only trained on 3 seconds long audio clips only. Yet, this was not the issue. When we looked at the spectrograms we found it very different from all the previous spectrograms that we worked with. One of those spectrograms is shown in figure 24.

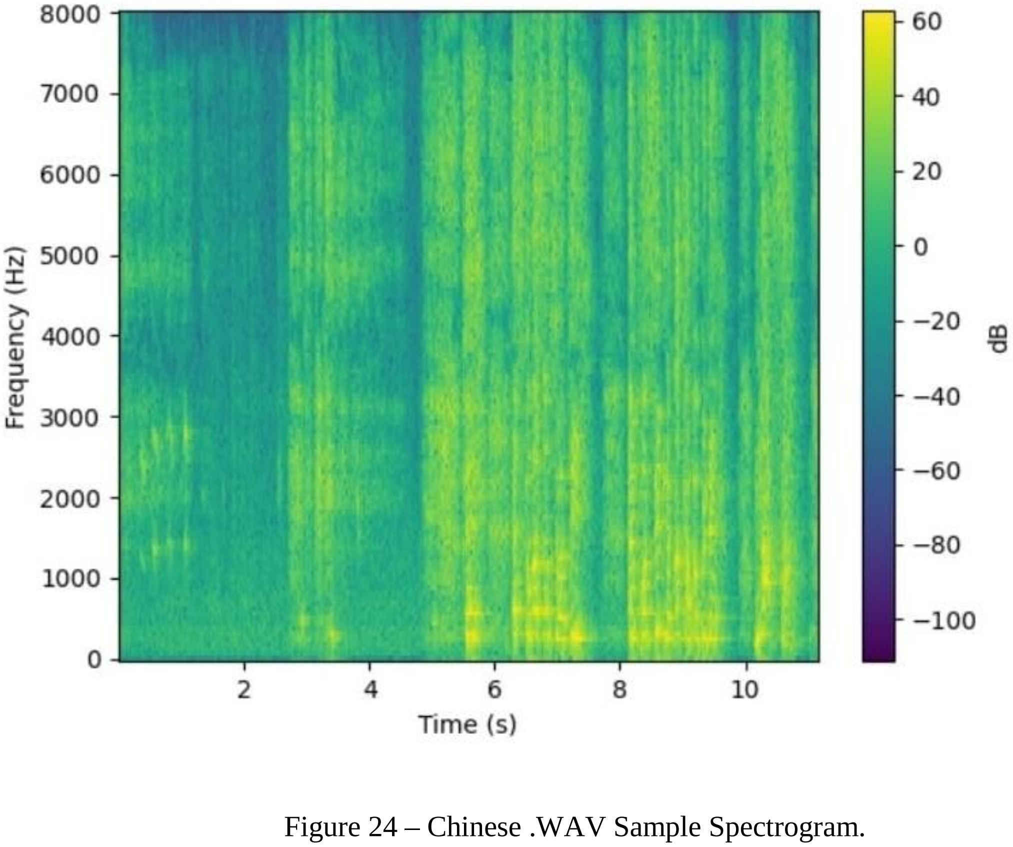


Figure 24 – Chinese .WAV Sample Spectrogram.

The issue was because of the data type. We trained our model on spectrograms that were generated from .mp4 files, while the ASVP-ESD data samples are .wav files. Therefore, we converted the data from wav format into mp4 and we tried again.

The model was able to recognize a lot of samples correctly! Including samples in Chinese, Russian, Spanish, and French. But what was even better is that it also worked on samples that are longer than 3 seconds! Figure 25 shows the spectrogram that was generated by a Chinese man who spoke for 9 seconds. His current emotion was happy and the model predicted so.

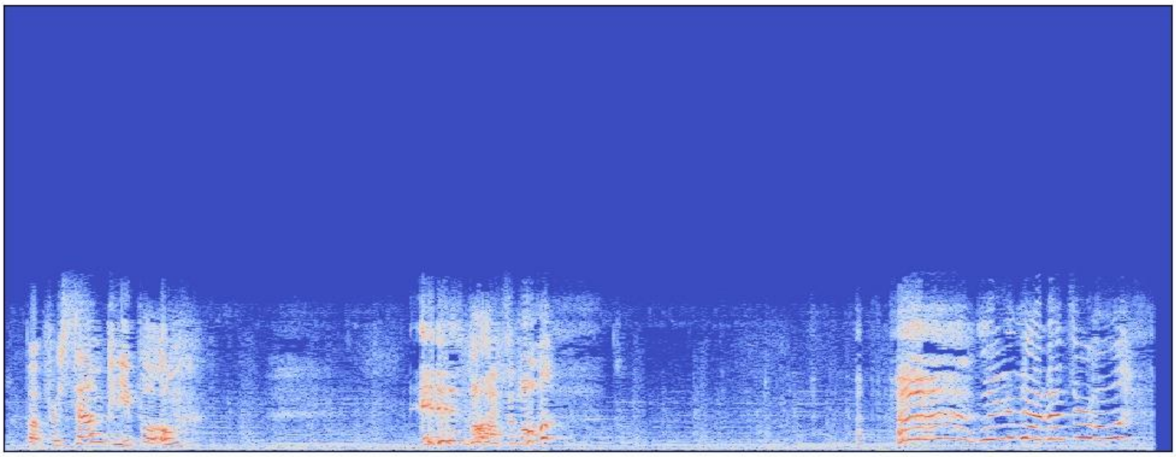


Figure 25 – Chinese Sample Spectrogram of Happy emotion.

4.3. Scenario

Let us say that Ahmad is a dedicated professional working in a demanding corporate environment. He often experiences high levels of stress and pressure due to tight deadlines, heavy workloads, and challenging projects. Ahmad recently discovered the "Relaxify" application, which enables him to record or upload audio clips to gain insights into his emotions and their relationship with stress. Here's how Ahmad can incorporate the application into his work routine:

1. Morning Reflection:

   Every morning, before starting his workday, Ahmad opens the "Relaxify" application on his smartphone. He finds a quiet corner in his office or takes a few minutes outside to record an audio clip using the application. By expressing his thoughts, emotions, and expectations for the day, he captures his initial emotional state which indicates his stress state.

2. Midday Check-In:
   As the day progresses and work-related pressures mount, Ahmad senses a surge of stress building up. Recognizing the need to assess his emotional well-being, he decides to utilize the "Relaxify" application during his lunch break. Ahmad records another audio clip, describing his current emotions and stress levels. This quick check-in helps him gain awareness of his stress triggers and provides a moment for self-reflection.

3. Regular Monitoring:
   Throughout the workday, Ahmad periodically checks in with the "Relaxify" application, especially during challenging or stressful moments. By recording audio clips, he monitors his emotional fluctuations and receives timely reminders to implement coping strategies. This enables him to maintain a healthier work-life balance, leading to improved productivity and job satisfaction.

Now that Ahmad can monitor his emotional and stress status throughout the working hours, he becomes aware of the specific times when he is most prone to experiencing stress. This newfound knowledge motivates Ahmad to develop a personalized coping strategy that he can implement at work. For example, he decides to practice deep breathing exercises during his short breaks or engage in a quick mindfulness session to alleviate stress. This will positively impact Ahmad's overall work performance. By managing his stress effectively, Ahmad is able to work more efficiently, make clearer decisions, and communicate more effectively with his colleagues. This, in turn, enhances the overall productivity and collaboration within the company or team, leading to improved work outcomes and a healthier work environment.

As Figure 26 demonstrates, the ability to track and analyze emotional states empowers users like Ahmad.

Figure 26 – Stress can be managed!

CHAPTER 5
CONCLUSIONS AND FUTURE WORK

In this section we conclude our project and discuss the future work that might be applied to our solution.

5.1 Conclusions

In conclusion, we successfully developed an AI solution for stress management in work environments. Our solution utilizes audio clips as input and transforms them into spectrograms to analyze voice characteristics and determine the user's current emotion. By leveraging this information, our model provides insights into the user's emotional state and predicts whether they are on the verge of experiencing stress.

The results of our AI solution are displayed to the user through a user-friendly application called “Relaxify” that we developed using Flutter. We also believe that our application is not limited to work environments alone; it can be applied in various environments including educational institutions and personal use. Therefore, we are grateful to say that we have successfully achieved the goal of our project.

5.2 Future Work

For future work, there are three main areas we would like to focus on to further improve our application. Firstly, we aim to enhance the application by transforming it into a real-time solution. Instead of requiring users to record or upload audio clips, the application will continuously analyze voice characteristics during work hours, providing timely notifications and insights to the user. This real-time approach will enable users to proactively manage their stress levels throughout the day.

Secondly, we plan to develop a database that stores stress records collected by the application. This database will serve as a valuable resource for generating weekly reports on the user's emotional state. By reviewing these reports, users can gain a better understanding of their stress patterns and take appropriate actions to improve their well-being.

Lastly, we intend to integrate speech-to-text functionality into the application. This addition will enable the application to analyze stress and emotional states more accurately, as voice characteristics can vary across different regions and languages. By leveraging speech-to-text technology, our application will be able to reach a wider audience and provide more precise insights into users' emotional well-being.

By focusing on these areas of improvement, we aim to enhance the effectiveness and usability of our application, ultimately empowering users to better manage their stress and emotional states in various contexts.

REFERENCES

[1] World Health Organization – WHO, "The impact of COVID-19 on mental health cannot be made light of ." Available:

https://www.who.int/news-room/feature-stories/detail/the-impact-of-covid-19-on-mental-health-cannot-be-made-light-of [Accessed January 2023].

[2] The National Institute for Occupational Safety and Health (NIOSH), "STRESS...At Work" Available:

https://www.cdc.gov/niosh/docs/99-101/default.html#:~:text=done%20about%20it.-,What%20Is%20Job%20Stress%3F,poor%20health%20and%20even%20injury. [Accessed January 2023].

[3] Better Health, “Anger - how it affects people” Available:

https://www.betterhealth.vic.gov.au/health/healthyliving/anger-how-it-affects-people [Accessed March 2023].

[4] HR Dive - "Worker stress costs employers billions in lost productivity". Available: https://tinyurl.com/3wbxx7aa
[Accessed January 2023].

[5] GitHub, by: soujanyaporia, Available:

https://github.com/declare-lab/MELD

[6] Kaggle, RAVDESS dataset, Available:

https://www.kaggle.com/datasets/uwrfkaggler/ravdess-emotional-speech-audio

[7] Kaggle, ANAD dataset, Available:

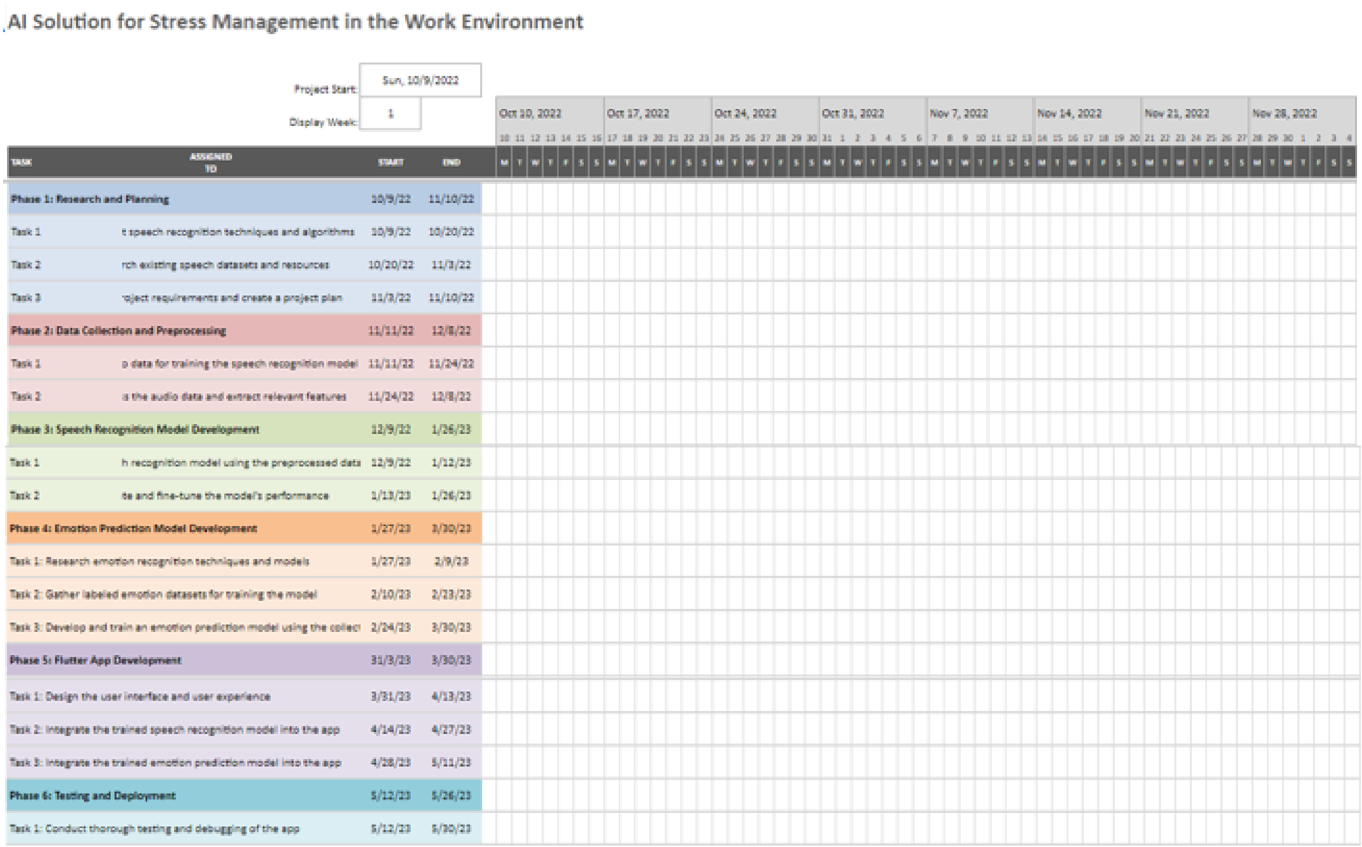
https://www.kaggle.com/datasets/suso172/arabic-natural-audio-dataset

[8] Kaggle, ASVP-ESD dataset, Available:

https://www.kaggle.com/datasets/dejolilandry/asvpesdspeech-nonspeech-emotional-utterances

APPENDICES

We used Microsoft Excel to generate our project time Gantt chart as shown in Figure 27.



AI Solution for Stress Management in the Work Environment

Project Start: Sun, 10/9/2022

Display Week: 1

TASK	ASSIGNED TO	START	END
Phase 1: Research and Planning		10/9/22	11/10/22
Task 1	t speech recognition techniques and algorithms	10/9/22	10/20/22
Task 2	rch existing speech datasets and resources	10/20/22	11/3/22
Task 3	'oject requirements and create a project plan	11/3/22	11/10/22
Phase 2: Data Collection and Preprocessing		11/11/22	12/8/22
Task 1	o data for training the speech recognition model	11/11/22	11/24/22
Task 2	s the audio data and extract relevant features	11/24/22	12/8/22
Phase 3: Speech Recognition Model Development		12/9/22	1/26/23
Task 1	h recognition model using the preprocessed data	12/9/22	1/12/23
Task 2	te and fine-tune the model's performance	1/13/23	1/26/23
Phase 4: Emotion Prediction Model Development		1/27/23	3/30/23
Task 1: Research emotion recognition techniques and models		1/27/23	2/9/23
Task 2: Gather labeled emotion datasets for training the model		2/10/23	2/23/23
Task 3: Develop and train an emotion prediction model using the collec		2/24/23	3/30/23
Phase 5: Flutter App Development		31/3/23	3/30/23
Task 1: Design the user interface and user experience		3/31/23	4/13/23
Task 2: Integrate the trained speech recognition model into the app		4/14/23	4/27/23
Task 3: Integrate the trained emotion prediction model into the app		4/28/23	5/11/23
Phase 6: Testing and Deployment		5/12/23	5/26/23
Task 1: Conduct thorough testing and debugging of the app		5/12/23	5/30/23

Figure 27 – Gantt Chart.