



**PROJECT TITLE: ANIMOJI EVOLUTION**  
**COURSE NAME: ADVANCED ALGORITHMS**  
**COURSE CODE: WOA 7001**

**GROUP MEMBERS:**

<b>Name:</b>	<b>Matric No:</b>
1. ZUHARABIH BT SULAIMAN	S2032539
2. A T M MANFAAT ZAYEEM	17221083/1
3. E Z E FLASH	S2001402

**LECTURER : DR. RAJA JAMILAH BT BINTI RAJA YUSOF**  
**SUBJECT : WOA7001 – ADVANCE ALGORITHM**  
**SUBMISSION DATE : 23 JUNE 2021**



**FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY**

Contents	
Project Introduction .....	3
Objective .....	4
Description .....	4
Team Roles .....	5
1. Project Analysis .....	6
2. Planning and execution based on FILA Form .....	7
DS1: Choose, define and analyze a set of emotions .....	10
Animoji analysis: Anger and Disgust .....	11
D2: Analyze, Choose and create the suitable Animoji to represent Anger and Disgust .....	15
Animoji Classification/grouping .....	18
D3. Using the Dynamic Time Warp (DTW) algorithm to detect Emotions .....	19
Speech recognition & Processing flow .....	19
Mel Frequency Cepstral Coefficients (MFCC Feature Extraction) .....	19
DTW Background .....	22
DTW algorithm .....	23
DTW Time Complexity .....	27
DTW Weakness/Disadvantages .....	27
Project Focus DTW implementation .....	27
Testing and Quality Checks .....	28
DS4: Create an interface or GUI for the above .....	37
Overall reflection of the experience doing the project .....	41
Conclusions .....	42
Appendix .....	43
References .....	46

## Project Introduction

This animated emoji group project is assigned to WOA 7001 students as part of final project assignment. Each team consist of 2 or 3 members. The students need to apply the learning outcomes which they have gained during the project phase. The learning outcomes are as below:

- Demonstrate familiarity with major advanced algorithms.
- Apply advanced design and analysis techniques.
- Apply concepts and skills to develop an information system
- Develop a speech/voice recorded message to identify and match the text input with anger and disgust Animoji using the Dynamic Time Wrapping (DTW) algorithm

Speech Recognition has gained significant popularity over the years due to the rapid advancement of Information Technology. As human speech emits emotions, recognition of various emotions has also gained popularity in the domain of Automatic Speech Recognition or ASR. Speech or voice can be simply defined as one dimensional signal. Like any other signals, it has properties like amplitude, pitch, frequency etc. This widely available technology can now easily be seen in various applications which can extract texts from the spoken words if they are clearly spoken. For example, in python, Google, IBM etc. has built-in libraries already, which are useful for implementing speech to text or STT.

Extraction of various emotions from speech has profound impact on research not only in the domain of Computing Linguistics, but also covering areas such as speech recognition in car parking (Kexin, T., et al., 2019), psychological domain such as anxiety, glossophobia etc. (El-Yamri, M., et.al., 2019). Gamified VR based research conducted by El-Yamri, M., et.al. (2019) developed an algorithm to measure the anxiety level of the speaker in a VR environment based on the speech given by the speaker. The research mainly focused on the speakers' voice tone. Thus, it can be claimed that the characteristic of speech has notable value in many domains.

In recent years, psychologists tried to segregate the complex emotion we have in our daily lives. A research conducted by Cowen, A. S., et. al. (2017) reports about 27 emotions that humans feel. Also, in recent times, an article written by Cherry, K. (2020) states six basic emotions namely, happiness, fear, anger, sadness, disgust, surprise. These basic emotions have also quantifiable characteristics in facial expressions. For example, happiness can be easily

identified by a smiling face, fear can be identified by widening of the eyes, Anger can be identified by glaring, disgust can be identified by the curling of upper lips (Cherry, K., 2020).

Animoji or animated emoji was first introduced electronic devices by apple in 2017. Animojis are animated emojis. Emojis were developed in Japanese mobile phones back in 1997 and thus gained popularity in the coming years (Blagdon, J., 2013). The reason of emoji getting popularized because of gap of emotional cues in conversation (Evans, V., 2017). As Animojis are animated, thus it can exert more emotions which fills up the emotional cues more. Speeches or voice are signals which has various quantifiable properties. The properties of voice can be calculated by MFCCs or Mel Frequency Cepstral Coefficients. Therefore, by the usage of this property, various emotional data can be extracted and matched (Likitha, M. S., 2017).

Dynamic Time warping or DTW is an algorithm to determine the similarity between two temporal signals. By combining the voice properties extracted by MFCCs and computing the temporal distance by DTW it is possible to develop a prototype of an Animoji determining system. In this project, emphasize will be given to the basic emotions, specifically Anger and Disgust. In the first phase of the project, the Animojis are created. Neutral and Unidentified Animojis are also considered because of the validation of the prototype.

## **Objective**

The objective of this project is to benefit students from

- a. Rapid project development and advance algorithm application, and
- b. To expose students in developing efficient software using specific algorithm design paradigm

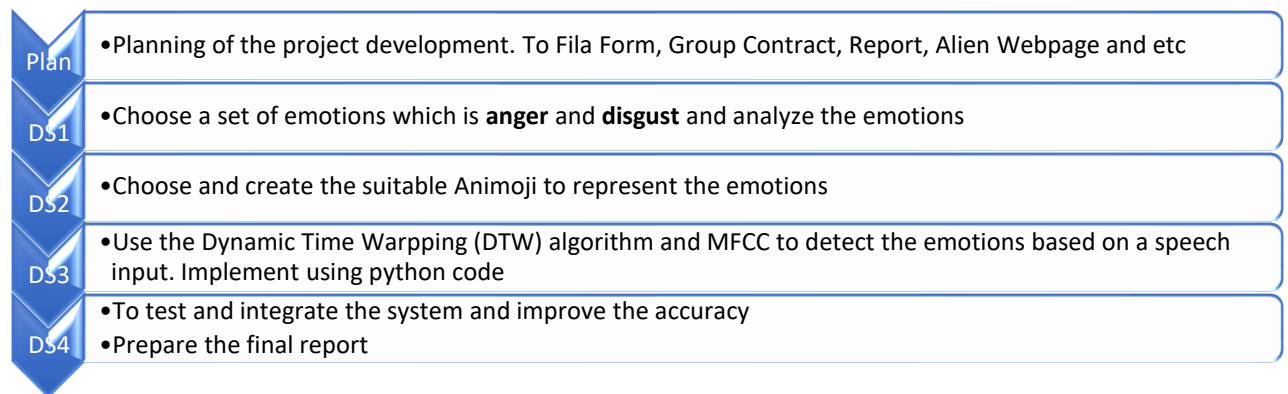
### **Learning Outcome involved for WOA7001:**

- a. Demonstrate familiarity with major advanced algorithms.
- b. Apply advanced design and analysis techniques.
- c. Apply concepts and skills to develop information system

## **Description**

The group project, which consists of three (3) members from Advance Algorithm subject need to implement an Animoji system that acts upon the type of speech sent in a recorded voice

message. The development period for the project is 4 weeks. The phase of the system implementation as shown in below flowchart.



High level Project timeline as per tabulated below, the project plan is designed with a target to be completed within 4 weeks period.

Week	Days	Period	Task Focus
1	7	28/05/2021 - 02/06/2021	<ul style="list-style-type: none"> <li>• Group contract alignment &amp; signing</li> <li>• Project Planning</li> <li>• (DS1) Analyze, choose &amp; define Animoji</li> <li>• (DS2) Analyze DS1 &amp; Create/Reuse Animoji</li> </ul>
2	7	03/06/2021- 09/06/2021	<ul style="list-style-type: none"> <li>• (DS3) Develop Algorithms using Dynamic Time Warp (DTW)</li> </ul>
3	7	10/06/2021- 16/06/2021	<ul style="list-style-type: none"> <li>• (DS3) Develop Algorithms using Dynamic Time Warp (DTW)</li> <li>• Program</li> </ul>
4	7	17/06/2021 - 23/06/2021	<ul style="list-style-type: none"> <li>• (DS3) Create an interface based on D3</li> <li>• (DS4) Code Integration</li> </ul>

### Team Roles

No	Matric No	Name	Team Role
1	S2032539	Zuharabih Sulaiman	SME Project Manager
2	17221083/1	A T M Manfaat Zayeem	Sr. Python Programmer
3	S2001402	Eze Flash Nwobia	IT Analyst

## 1. Project Analysis

Based on the requirement given by Dr Raja Jamilah. The project is to detect 2 emotions for group 1 which is anger and disgust from voice. Hence, in order to achieve the objective to detect the emotion we have gone through a process which involves all the steps mentioned below. The process to develop a working program that detects emotion and gives an Animoji output has to be planned and executed in an organized manner given the project timeline is 4 weeks and with limited number of resources (team members) assigned to the project. The steps are listed in a sequence as below:

1. Analyze the project requirement
2. Analyze the scope and limit the project scope upon defining limitation (such as time and resources)
3. Assess team members strength and weakness
4. Discuss and brainstorm all team member's understanding regarding the project requirement and scope
5. Discuss and agree on project scope
6. Outline and agree on group contract to define the ways of working within the team
7. Define and agree on team member's role and responsibilities
8. Work on individual task and align with team members in a daily stand-up call and technical meetings
9. Technical problem solution
  - Install PyCharm in all team members laptop
  - Assess speech to text python implementation and libraries
  - Assess voice recognition methods and functions in python
  - Assess MFCC feature extraction concept in python
  - Assess MFCC library and functions
  - Assess DTW functionalities and algorithm and how it can be implemented in speech recognition
  - Assess DTW function and libraries in Python
  - Assess anger and disgust emotion threshold values in terms of time warping and distance calculation
  - Install all necessary packages to be used in the program
  - Implement speech to text functions from speech recognition library

- Implement MFCC feature extraction into training voice and input MFCC features to calculate DTW distance between the 2 voices
- Calculate distance on training voice and input voice using MFCC feature extracted
- Test and define distance threshold values acceptable for 3 levels of anger, 3 levels of disgust and neutral emotions
- Test and improve the accuracy of emotion detection from the voices

## 2. Planning and execution based on FILA Form

In Week 1 we discussed out the overall project planning, defined and updated group contract, defined the project planning, selected the emotion for the project, created a high level Animoji program design process flow, and completed DS1 and DS2 sections. There is also other subtasks in DS1 and DS2 split among the team members to work and some of DS3 and D4 studies were also done to be able to estimate the future task accurately.

FACTS	IDEAS	LEARNING ISSUES	ACTION	DATELINE
What we know about the task	What do we need to find out?		Who is going to do it?	02/06/2021
Planning	Define & Update group contract		Zuha	02/06/2021
	Define project planning		Zuha/Zayeem/Flash	02/06/2021
	Select emotion for the project		Zuha/Zayeem/Flash	02/06/2021
	Animoji program design process flow		Zuha/ Zayeem/Flash	02/06/2021
	Consolidate content in the report		Zuha/ Zayeem	02/06/2021
	Create FILA form		Zuha	02/06/2021
DS1: Define& Analyze Animoji	Introduction		Zayeem	02/06/2021
	Project background/Literature Review		Zuha/Zayeem	02/06/2021
	Animoji Analysis		Zuha/Zayeem/Flash	02/06/2021
	References		Zayeem	02/06/2021
DS2: Create Animoji	Create Anger, Disgust, Neutral, Unidentified Memoji		Zuha	02/06/2021
	Record Memoji Anger, Disgust, Neutral, and Unidentified		Zuha	02/06/2021
	Convert Memoji video into Animoji GIF		Zuha	02/06/2021
DS3: DTW Algorithm	Research on DTW		Zuha/Zayeem/Flash	02/06/2021
DS4: Program coding	Research on existing speech recognition program		Zuha/Zayeem/Flash	02/06/2021

In week 2 we continued the project task moving on into DS3 to implement and make sure to have a working DTW algorithm implementation in python. Along with that we also work in improving DS1 and DS2 based on week 1 feedback from Dr. Raja Jamilah. The rest of the task details are as shown in the table below.

FACTS	IDEAS	LEARNING ISSUES	ACTION	DATELINE
What we know about the task	What do we need to find out?		Who is going to do it?	09/06/2021
Planning	Project week 2 task planning		Zuha	09/06/2021
	Create/Update FILA form		Zuha	09/06/2021
	Upload progress file on Github		Zuha	09/06/2021
	Create website (Front end)		Flash	09/06/2021
	Upload and create links for team's progress		Flash	09/06/2021
DS1: Define& Analyze Animoji	Animoji Analysis Update		Flash	09/06/2021
DS2: Create Animoji	Update and convert Animoji into GIF		Zuha	09/06/2021
DS3: DTW Algorithm	Report: Speech Recognition and process flow		Zuha/Zayeem	09/06/2021
	Report: MFCC		Zayeem	09/06/2021
	Report: DTW Background		Zuha/Zayeem	09/06/2021
	Report: DTW Algorithm Pseudocode		Zuha/Zayeem	09/06/2021
	Report: DTW Time Complexity /Weakness		Zuha	09/06/2021
	Report: Project Scope limitation DTW Implementation		Zuha/Zayeem	09/06/2021
	Python code: implementation (part 1 – voice to text input)		Zayeem	09/06/2021
	Python code: implementation (part 2 – Animoji GIF output)		Zuha	09/06/2021
	Python code: implementation (part 3 – detecting the pitch/frequency/amplitude)		Zuha/Zayeem	09/06/2021
	Python code: implementation (part4 – implementing DTW part1)		Zayeem	09/06/2021
	Python code: implementation (part5 – implementing DTW part2)		Zuha	09/06/2021
	Testing: Python program testing and quality control		Flash	09/06/2021
	Testing: Python speech recognition testing method		Flash	09/06/2021
DS4: Program integration	Continue research on existing speech recognition program using DTW		Zuha/Zayeem/Flash	09/06/2021

In week 3 we continued with DS4 implementation and continued working on DS3 improvement to ensure emotion detection works with higher accuracy as the result in week3 was not measurable due to lack of time for regression testing.

FACTS	IDEAS	LEARNING ISSUES	ACTION	DATELINE
What we know about the task	What do we need to find out?		Who is going to do it?	16/06/2021
Planning	Project week 3 task planning		Zayeem	16/06/2021
	Create/Update FILA form		Zuha	16/06/2021
	Upload progress file on Github		Zuha	16/06/2021
	Update website (Front end)		Flash	16/06/2021
	Upload and create links for team's progress on website		Flash	16/06/2021
DS1: Define& Analyze Animoji	Animoji Analysis Update		Zayeem	16/06/2021
DS2: Create Animoji	Update and convert Animoji into GIF		Zuha	16/06/2021
DS3: DTW Algorithm	Report: MFCC		Zayeem	16/06/2021
	Report: DTW Algorithm Pseudocode		Zuha/Zayeem	16/06/2021
	Report: DTW Time Complexity /Weakness		Zuha	16/06/2021
	Report: Project Scope limitation DTW Implementation		Zuha	16/06/2021
DS4: Program integration	Python code: Code optimization		Zuha/Zayeem	16/06/2021
	Python code: GUI improvement (part1- input)		Zuha	16/06/2021



	Python code: GUI improvement (part2 – output consolidation)	Zayeem	16/06/2021
	Python code: Implementation update to improve Animoji output	Zuha	16/06/2021
	Python code: Explore how to make python program to exe	Zuha	16/06/2021
	Python code: Explore standard deviation calculation to improve accuracy	Zayeem	16/06/2021
	Testing: Testing the input and output Animoji	Zuha/Zayeem/Flash	16/06/2021

In week 4 we worked on the final report writing and presentation slide preparation. And we've distributed the task of presentation slide preparation accordingly and had a final meeting 1 day before the presentation day and wrapped up the deck content.

FACTS	IDEAS	LEARNING ISSUES	ACTION	DATELINE
What we know about the task	What do we need to find out?		Who is going to do it?	23/06/2021
Planning	Project week 4 task planning		Zuha	23/06/2021
	Create/Update FILA form		Zuha	23/06/2021
	Upload progress file on Github		Zuha	23/06/2021
	Upload and create links for team's progress on website		Flash	23/06/2021
DS3: DTW Algorithm	Report Writing: Overall report structure writing		Zuha/Zayeem	23/06/2021
	Report Writing: Part1		Zuha	23/06/2021
	Report Writing: Part 2		Zayeem	23/06/2021
	Report consolidation and finalization		Zuha/Zayeem	23/06/2021
DS4: Program integration	Python code: Code optimization – Only DTW distance for detection		Zuha	23/06/2021
	Python code: Code optimization – DTW & Std Deviation		Zuha	23/06/2021
	Python code: Code Labelling		Zuha	23/06/2021
	Python Code: Print distance on output GUI		Zayeem	23/06/2021
	Testing: Only DTW distance		Zuha/Zayeem/Flash	23/06/2021
	Testing: DTW distance & word		Zuha/Zayeem/Flash	23/06/2021
	Testing: DTW & Std Deviation		Zuha/Zayeem/Flash	23/06/2021
	Testing: Diff test scenarios		Zuha/Zayeem/Flash	23/06/2021
Presentation Slide	Slide structure creation and planning		Zuha	23/06/2021
	Slide preparation Part 1		Zuha	23/06/2021
	Slide preparation Part 2		Zayeem	23/06/2021
	Slide preparation Part 3		Flash	23/06/2021
	Slide finalization		Zuha	23/06/2021

## Animoji program design overview

# Program Development Process Flow

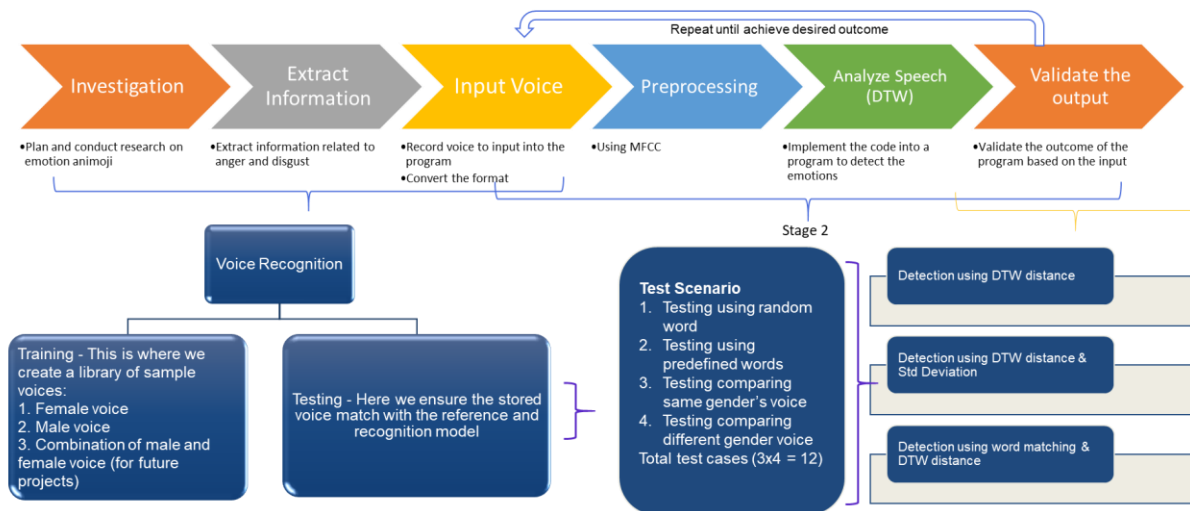


Figure1: High level Program development flow

## DS1: Choose, define and analyze a set of emotions

There are different types of emotions, and they can be expressed in many ways, both verbal and facial. Some simple expressions of emotions are given as below:

- **Anger:** violence, hostility, resentment, wrath, irritability, fury, and outrage.
- **Shame:** regret, guilt, contrition, chagrin, remorse, and embarrassment.
- **Sadness:** depression, grief, melancholy, gloom, despair, sorrow, and loneliness.
- **Disgust:** scorn, contempt, distaste, disdain, revulsion, and aversion
- **Fear:** anxiety, fright, nervousness, dread, apprehension, and panic.
- **Surprise:** wonder, amazement, astonishment, astound, and shock.
- **Joy:** enjoyment, thrill, delight, bliss, relief, pride, happiness, and ecstasy.
- **Interest:** devotion, acceptance, affection, trust, kindness, love, and friendliness.

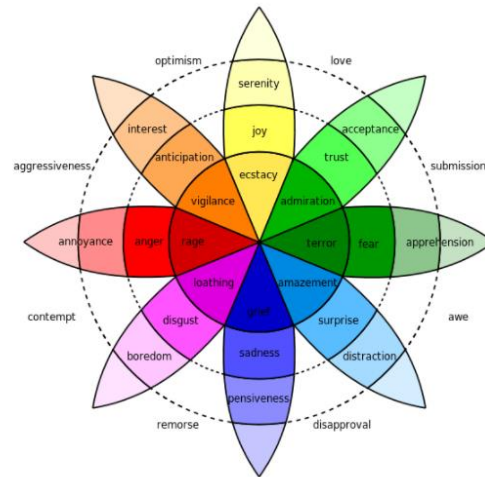


Figure2: Emotion levels

### **Animoji analysis: Anger and Disgust**

To design the system, each group has to implement an Animoji system that acts upon the type of speech sent in a recorded voice message. With the purpose to design a good algorithm, each group was required to choose a set of emotions and work on those particular emotions. Our group agreed and choose Set 1 out of 3 options which is anger and disgust

The study of emotions is greatly aided by digital signal processing hardware and software. However, machines cannot equal the performance of human equivalents in terms of accuracy and speed, especially with regards to speaker-independent emotion recognition. There are namely two phases of emotion recognition algorithms which are the testing phase and training phase.

#### **Anger Animoji**

Anger may sound as a simple expression but rather it is a complex expression of human expression. This emotion can be influenced by various factors, and it can also make an individual go beyond control (Seunagal, G., 2021).

The image of a face with furrowed eyebrows and with its mouth curling downward is the emoji representing anger, upset or disapproval. It is typically used to emphasize that someone is upset or furious. Angry Face Emoji can mean “I am so upset right now!” or “This thing angers me so much! I hate it!”. The Angry Face Emoji appeared in 2010, and also known as the Mad Face. Sometimes it is mentioned as the Mad Emoji. Figure 1 Below is pictorial representation of each organization with their anger emoji.

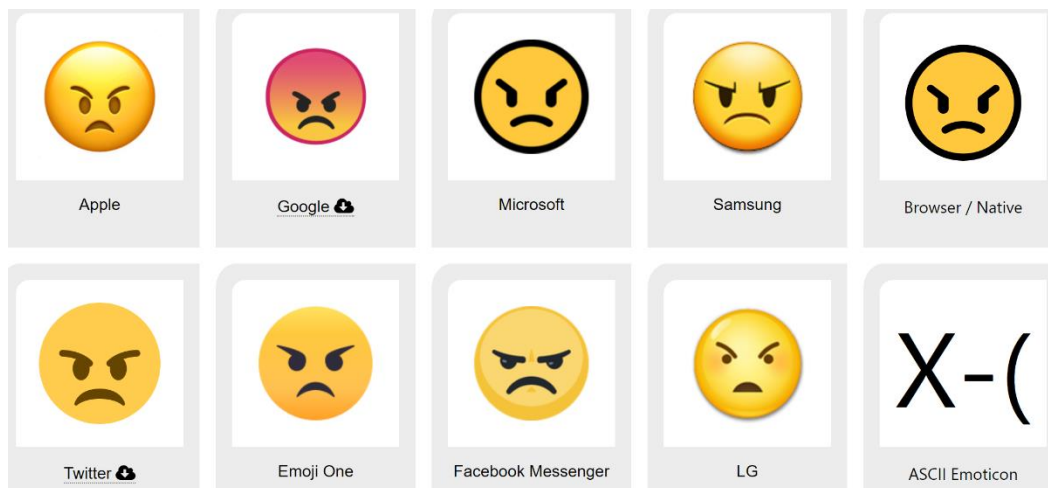


Figure 3: Anger Emoji

### Levels of anger:

According to Seunagal, G. (2021), who is profound writer in the field of mental health and psychological therapy, defined anger in four levels. They are listed below, and short definition is given.

#### 1<sup>st</sup> level – Annoyance:

Annoyance can be defined as the first level of anger. It subsides easily compared to other levels of anger. An individual can be annoyed by small things though this topic can be subjective. For example: traffic jams, hearing constantly to someone making unnecessary sounds etc. This level of anger can be regulated easily.

#### 2<sup>nd</sup> Level – Frustration:

If annoyance lasts longer for an individual, it goes to the second level of anger, which is frustration. It also impacts on the individual's concentration level. It is also linked to negative thinking or emotional state. It takes longer to go away compared to annoyance.

#### 3<sup>rd</sup> Level – Hostility:

The third level of anger is Hostility. It appears to an individual when someone is constantly or consistently exposed to displeasing or threatening situations. An expression of hostility is screaming. This level of anger leads to the next level which is called Rage.

#### 4<sup>th</sup> Level – Rage:

In this level of anger an individual can throw objects, threat someone. It is also possible for an individual to get physical. It is considered dangerous because people are out of their mind and unable to control the situation. In other words, it can be said that anger has taken over the individual.

Particularly, in this course project, emphasis will be given to up to third level of anger, namely, Annoyance (Anger level 1), Frustration (Anger level 2), Hostility (Anger level 3). This is because, the project will only deal with the verbal speech, therefore, it is not possible to detect visual anger.

### **Disgust Animoji**

Ugh, nasty! That's the sentiment (and appearance) of the face with open mouth vomiting emoji, used to express literal and metaphorical disgust. Related words that are synonymous to disgust emoji are listed below

- 🤮 Spew
- 🤮 Throwing Up
- 🤮 Vomit

Figure 2 Below is pictorial representation of each organization with their disgust emoji

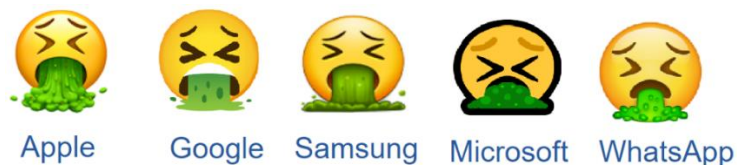


Figure 4: Disgust Emoji

### **Levels of disgust:**

According to Ekman, P. (2021) there are different intensities of disgust. Namely – DISLIKE, AVERSION, DISTASTE, REPUGNANCE, REVULSION, ABHORRENCE, LOATHING. DISLIKE being the least intense and LOATHING being the highest level of disgust. The different intensities of disgust as described by Pail Ekman can be depicted as below (Ekman, P., 2021):

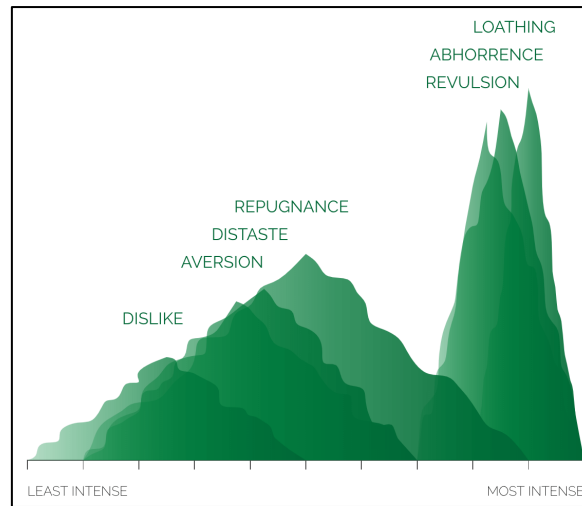


Figure 5: Different intensities of Disgust. (source: Atlas of emotions)

For all the different levels of anger and disgust above voices are represented with higher or lower sound amplitude as compared to a neutral emotion's voice. In terms of vocal emotional recognition, each level of emotions will be represented by different tone, pitch, amplitude, and frequencies. Hence normal sound volume of angry voices might not be enough to elicit differential emotional responses to angry tonality (Simon et al, 2016). Acoustic cues such as loudness help in deciphering spoken emotion. Sound intensity can have a quantifiable effect on the emotional impact of the sounds that it contributes to. Thus, sound intensity should not be treated as a simple control and should be included for vocal emotion investigations. (Chen et al, 2012).

## **D2: Analyze, Choose and create the suitable Animoji to represent Anger and Disgust.**

Advancement of neuropsychology has opened the opportunity to explore various emotions in humans. There are significant number of studies which talks about the six basic types of human emotions (Cowen. 2017). According to Cherry, K. (2020), there are different types of emotions that have influences on how we interact with others. She stated that, during the 1970s psychologist Paul Eckman identified six basic emotions which are universally experienced in all cultures. These basic emotions have specific facial expressions as shown in Table 1 below.

Table 1: Emotion analysis

Happiness	Fear	Anger	Sadness	Disgust	Surprise
Smiling	Widening the eyes and pulling back the chin	Frowning or glaring	Crying/ Quietness/ Dampened mood	Wrinkling the nose and curling the upper lip	Raising the brows, widening the eyes, and opening the mouth

For this project, **Anger** and **disgust** are chosen. As there are different level of emotions are exerted by an individual in certain circumstances, the Animoji that are designed for this project tried to follow the research and article of the above-mentioned published materials. Besides, there are also two Animoji's are designed. They are – *Neutral* and *Unidentified*.

### **Animoji Creation**

As per explained in the introduction above, we have created total of 8 Animoji which includes three levels of anger, three level of disgust emotions, one neutral and one unidentified Animoji. This is due the scope of project that only covers anger and disgust emotion leaving the rest of the emotions matched into unidentified emotion.

### **Animoji created and saved in MP4 format**

In order to create an Animoji for this project, we have decided the approach to use Memoji in IPAD and create a video to show emotions expression and it is saved in MP4 format. This will the be converted into GIF format as shown in the next sections.

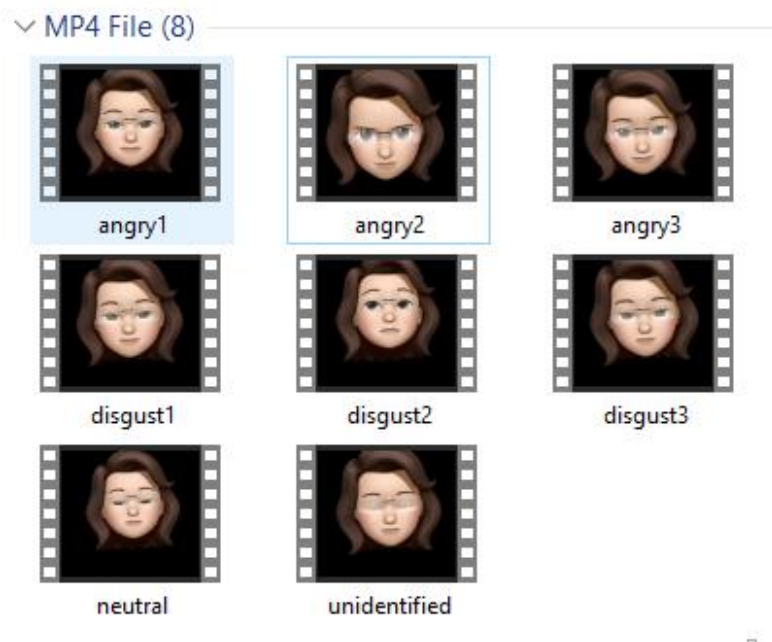


Figure 6: MP4 Memoji Videos

### Animoji converted into GIF format

All the Memoji videos in the previous section is converted into GIF as shown in below figure accordingly. All the MP4 are converted into GIF using an online tool in <https://ezgif.com/> website.

File size: 42.38KiB, width: 432px, height: 320px, type: mp4 (video), length: 00:00:02 [convert](#)

Notice: video preview may have reduced quality, but it won't affect the GIF.  
Not all types of video can be played with this player and/or web browser, so if the player is not working for you, you can try to generate the GIF anyway.

If you intend to crop part of the image, it's better to crop the video before converting it to GIF, smaller video might produce better quality GIF.

Start time (seconds):  [Use current position](#)

End time (seconds):  [Use current position](#)

Size:

Frame rate (FPS):

Method:

☐ Optimize for static background (assign more colors to moving parts of the image)

[Convert to GIF!](#)

Figure 7: Animoji Creation



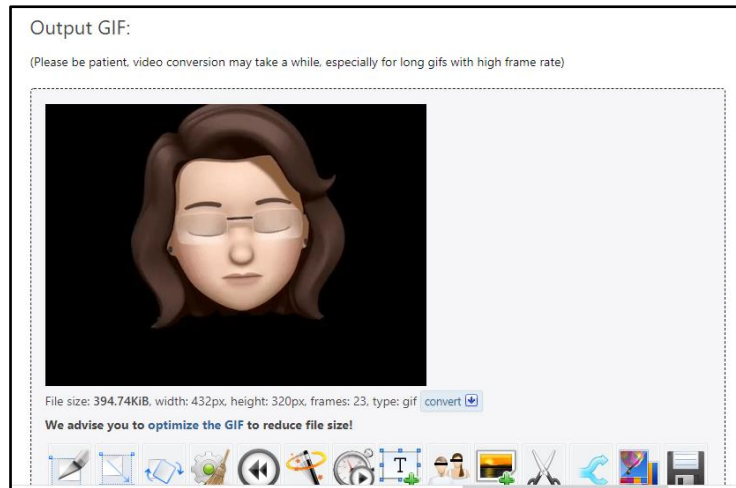


Figure 7a: Animoji Creation

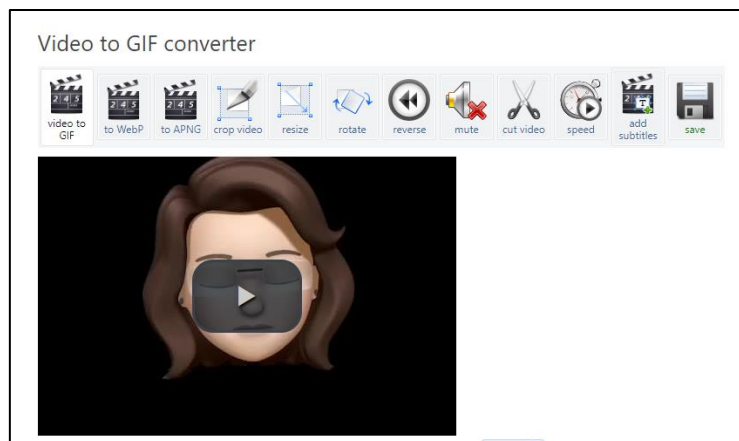


Figure 7b: Animoji Creation

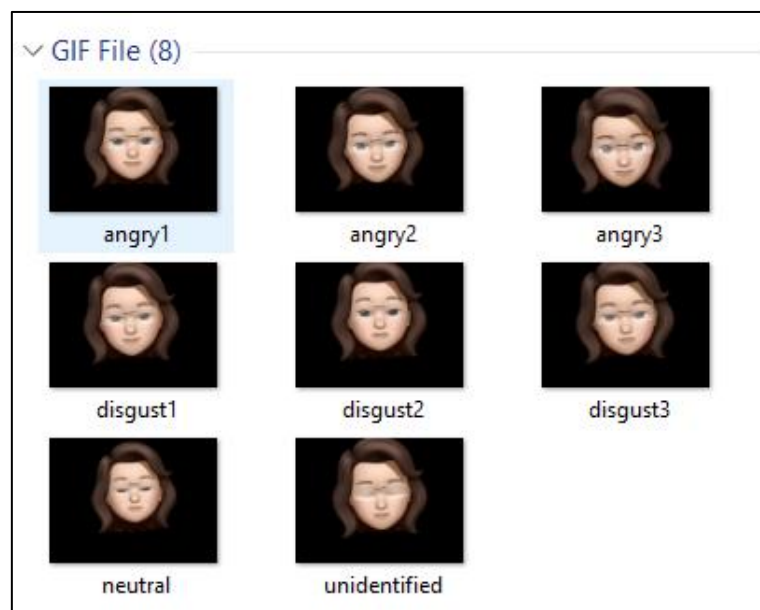










Figure 7c: Animoji Creation

## Animoji Classification/grouping

Table 2: Animoji for all emotions selected for this project

LEVEL EMOTION	Lightly affected	Medium affected	Highly affected
Anger			
Disgust			
Neutral			
Unidentified			

### D3. Using the Dynamic Time Warp (DTW) algorithm to detect Emotions

#### Speech recognition & Processing flow



Figure 8: Speech Recognition processing steps

In order to find out the emotion in the received voice signal, we need to have a large amount of information like energy, power spectral density (PSD) and so on in order to perform a statistical analysis. MFCC is enabling us to reach this goal. Also, to eliminate the disparity between the database and the input signal, we employ a method called Dynamic Time Warping (DTW).

An analysis of the user's voice is performed following the process of getting an input through a microphone. Manipulation of the input audio stream is an integral part of the architecture of the system. Additionally, at various levels, various processes are applied to the input signal such as Pre-emphasis, Framing, Windowing, and Mel-Cepstrum analysis, all of which have the objective of recognising (matching) words spoken. The voice algorithms are made out of two distinct steps. In the first case, it is referred to as training, whereas in the second case, it is commonly known as a mission rehearsal or a mission exercise.

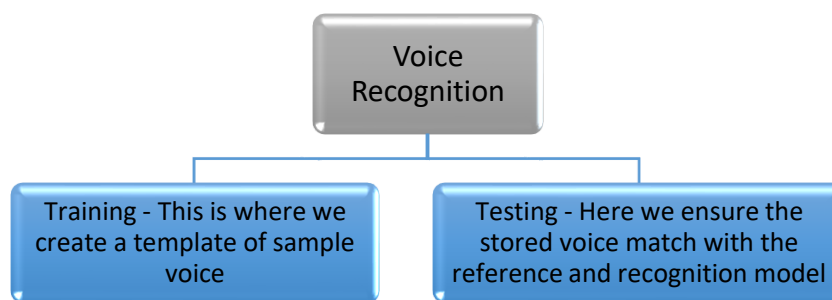


Figure 9: Voice recognition main steps

#### Mel Frequency Cepstral Coefficients (MFCC Feature Extraction)

MFCC or Mel Frequency Cepstral Coefficients, is basically the coefficients of the ‘Cepstrum’ of the audio signal. The usage of MFCC features dates way back in 1960s and is

also used in music processing. Mel-frequency cepstral coefficients (MFCCs) are the coefficients which make up an MFC collectively for the signal.

To begin with, how the **Cepstral** part of the MFCC needs to be discussed first. The signal in the time domain ( $x(t)$ ), which is a normal wave form. Then discrete Fourier transform is applied to the signal. Upon getting the result, a spectrum is generated for the for the signal. This operation takes the signal from time domain to frequency domain. Then logarithm is applied to the whole result which gives the logarithmic amplitude spectrum of the signal. After that, inverse Fourier transform is applied to the logarithmic amplitude signal. In short, it can be said that, calculating the cepstrum is first applying the discrete Fourier transform and then taking the inverse Fourier transform of the result. As we first calculating a spectrum and then again calculating a spectrum by taking the inverse Fourier transform, the name **cepstrum** comes from this technique of mathematical operations. It can be said that Cepstrum is a spectrum of a spectrum of signal. In equation, it can be showed as below:

$$C(x(t)) = F^{-1} [\log (F[x(t)])]$$

The original signal is represented in time domain ( $x(t)$ ). When DFT or discrete Fourier transform is applied, it transforms the signal to frequency domain. Applying inverse Fourier transform means transforming a frequency domain to another frequency domain, which leads to Quefrency. The figure below, gives an idea of *cepstrum* and *quefrency*.

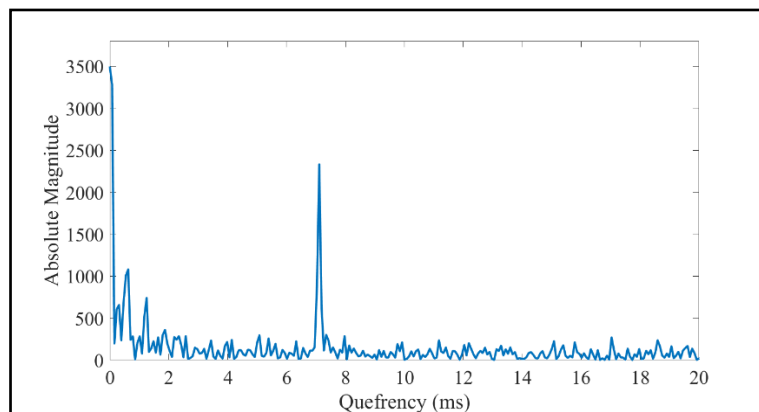


Figure 10: Depiction of Quefrency by applying IFT to the power spectrum

### Reason behind using MFCC is speech analysis domain:

To formalize a speech signal, it can be defined as the convolution of the response of vocal tract frequency with glottal pulse (Velardo, V., 2020). By applying logarithm operation, the two components are separated. Only the vocal parts of the signal are important while

extracting information from a speech. Most of the importance parts in the speech, comes from the vocal part. MFCC separates the glottal part of the speech signal and brings out the important vocal part of the speech signal such as *formants*, *phonemes*, *timbre* etc. (Wolfe, J., n.d.). This is the advantage of going from frequency domain to quefrency domain which eliminates the glottal part. According to Velardo, V. (2020), the steps of retrieving MFCC from a speech signal is as below:

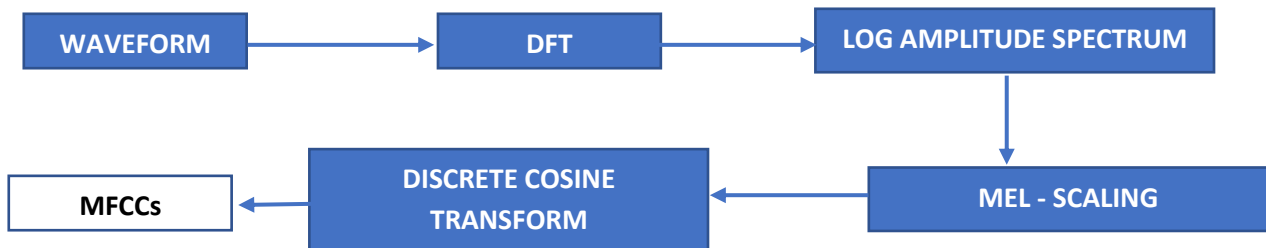


Figure 11: Flowchart of retrieving MFCCs from speech signal

There are traditionally 12-13 coefficients which are used to retrieve information form speech signals. In this project to find the anger and disgust emotion in the speech, mean of the MFCC values are taken into consideration, which was also done by another research by Likitha, M. et al. (2017). In python, Librosa library has the functionalities to extract MFCC values for the recorded speech signal. The library was used while doing the project.

The Figure below shows the mean of MFCC values calculated for one of the voices recorded in the project. The output we see in the array below is a time series extracted from the voice file which will later be used for the time warping and distance comparison to identify the emotion.

```

(venv) C:\Users\Zuha\PycharmProjects\AnimojiEvolution>python source.py
[-4.4876694e+02  1.0109952e+02 -3.3746083e+00  3.1109591e+01
 -1.0717710e+00  1.1564893e+01  4.8446374e+00  7.0536180e+00
 -1.6756659e+00  1.2610603e+01 -4.1651793e+00  2.3055232e-01
 -7.1498160e+00 -9.5813398e+00 -3.1372120e+00 -7.0607796e+00
 -8.7929744e-01 -3.6327257e+00  7.1240294e-01 -3.3427670e+00
 -2.4896905e+00 -1.3634344e+00 -1.3618975e+00 -5.1387339e+00
 -4.4572177e+00 -3.9109864e+00 -6.3555675e+00  1.7924280e+00
  5.0221887e+00  2.6576712e+00  8.6756306e+00  5.2579322e+00
 -8.6216366e-01 -3.1181982e+00 -4.5208230e+00 -2.8624134e+00
  2.9288077e-01 -4.1277635e-01 -2.2548361e+00 -3.9254510e+00]
  
```

Figure 12: Mean of MFCC from librosa.feature.mfcc() function

## DTW Background

Dynamic Time Warping (DTW) is an approach identify the lowest distance of two signals and gives a metric to compare the compatibility of matching up to the reference signal, in this particular project speech emotion recognition. DTW is a pattern recognition method to compare different time series/ signals/ time zones. The closer the two sounds are, the more alike they are. Thus, both sound patterns can be said to be same. The preliminary speech recognition data is converted into frequencies. The distance of sound around the recording is affected by volume, pronunciation time, and noise. The smaller the effect, the smaller the distance.

An audio signal produces a time series. DTW projects each element in the series onto the temporal dimension. As such, DTW finds the optimum distance (i.e., the shortest route) while completing this mapping. Time will be warped. This table contains the distances between locations, which is called *Memoization*. When the shortest paths are calculated, a similarity metric between two time series is generated. As all of this happens, a warping path is made. Because the path has been altered, the two series have the same time levels. when the warp path shrinks, the similarity between the two-time series grows Warping paths have rules. Warping is defined by these rules. Warping is done to both series.

The DTW algorithm is a traditional algorithm that is easy to carry out, making it useful in speech recognition. The DTW algorithm uses the point matching method to determine the matching distance. At higher volumes, the reference template and test voice require more time to match, hence recognition times will increase. Feature extraction cannot be done directly because of the non-stationarity of the speech, as well as external noise. After pre-processing, the voice signal is used to extract distinctive parameters.

Signal characteristics metrics such as short time energy, short-term zero-crossing rate, and short-time autocorrelation coefficient are abundant. Since LPCC's speech parameters extraction is quite accurate, the computation speed is comparatively rapid and hardware-based. But LPCC in anti-noise performance, robustness, and the recognition rate and other aspects are below average, thus in practise, MFCC is utilised. The input voice is used to extract parameters, and the parameters are stored in a library of reference templates. Recognition speech's feature parameters are obtained in the same way and utilised to match with reference template library

using the DTW algorithm, which yields the maximum similarity reference template in the library. DTW has been a very successful recognition matching algorithm.

### 3.3.1 Time Warping calculation and chart between 2 voices

The time warping graph below is generated from 2 voices where the bottom graph refers to the training voice and the upper graph shows the input voice. The orange lines shows the best alignment between the 2 time series. It shows that it ignores shifts in time dimension. It also ignores the speeds of the 2 time series. It is ahead of Euclidean distance that does point to point comparison which is hard to reach towards accuracy considering the time shifts that may occur at any point of time.

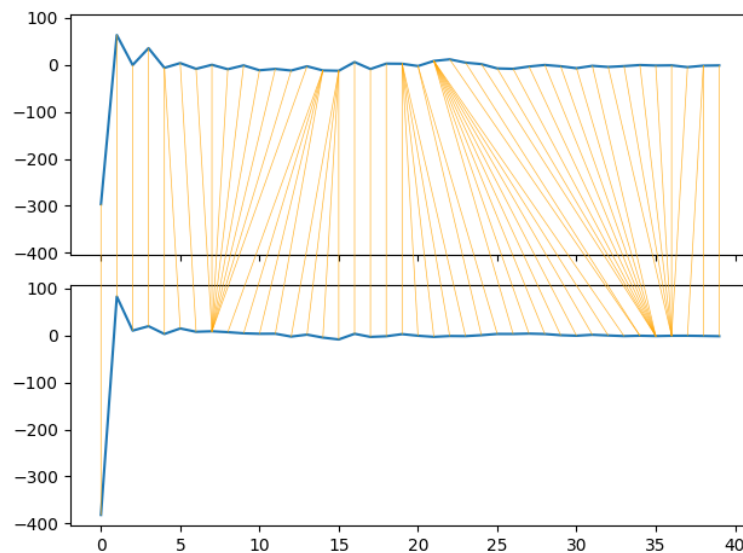


Figure 13: Dynamic Time Warping Graph

### DTW algorithm

In a published journal by Fong, S. (2012), the pseudocode of Dynamic Time Warping can be written as below:

```
DTW ( $v_1, v_2$ ) {
    // Vector  $v_1 = (a_1, \dots, a_n)$ ,  $v_2 = (b_1, \dots, b_m)$  are the time series
    // with n and m points
    // Let a matrix of 2-dimension S for similarity measure.
    S[0,0] = 0
    for (i=1 to m)
        S[0, i] =  $\infty$ 
    for ( i=1 to n)
        S[i, 0] =  $\infty$ 
```

```

//After initialising infinity value to all the indices, fill the
//similarity matrix with the difference of the two vectors
for (I =1 to n)
    for (j=1 to m) {
        cost = d (v1, v2)
        // Difference between time series points
        S [ i , j ] = cost + MIN ( S[i-1, j ],
                                   S[I, j-1],
                                   S[i-1, j-1] )
    } // End of inner for loop
} // End of outer for loop
Return S[n,m]
}

```

For example, the speech signal that is being used as the reference for comparing to the recorded or some other speech signal, comprises of M number of frames. These M number of frames are represented as vectors. For simplicity, the vectors can be represented as below:

**{R (1), R (2), R (3), ..., R (m), ..., R(M)}**

To generalize the above sequence of vectors, the above sequence of vectors can be written as below:

**R(I) [ Where, I = 1,2,3 ..., M]**

As Dynamic Time Warping – DTW algorithm determines the closeness between two time series or signals, there will be another test speech signal which will be compared to the reference vectors or signal. The testing signal can also be represented as vectors like the reference signal. Considering the test signal has N number of frames, the test vector's representation can be shown as below:

**{T (1), T (2), T (3), ..., T (n), ..., T (N)}**

The above sequence of vectors can also be represented in general as below:

**T(I) [Where, I = 1,2,3, ..., N]**

In general words, two signals or two time series can be said to be similar, if their distance is small or in some situation same or follows a same distance pattern. This distance measuring mechanism can easily be illustrated by measuring the distance on each pair of frames in both the signals. This frame wise distance measuring can also be called Euclidean distance measuring or point to point measuring. This can be one way to find the similarity between two time series or signals. But this technique has its limitations when there are some distortions in the testing signal but at the same time, it has similarity with the reference signal after a certain period. In those scenarios, Euclidean distance measuring technique will fail and thus will result in inaccurate results. The above scenario can be depicted as below:



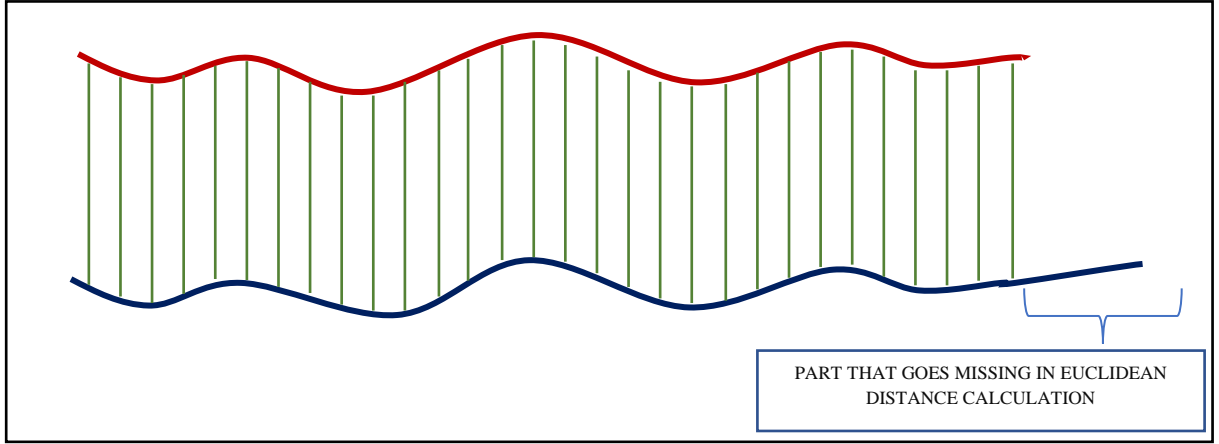


Figure 14: Depiction of Euclidean Calculation of two time series

In the above depiction, point to point Euclidean distance calculation is unable to measure the extended part of the blue labelled signal. It might also be the case that the left out blue signal might have carried vital information of distance. DTW is a dynamic approach in calculating the distance of two signals. It contains the ability to measure distance from one point to many points. Therefore, even if there are many peaks and troughs, the algorithm tries to find the similarity by measuring from one point to many points. As a result, no curves are left out (Zhang, J., 2020). To explain further about DTW algorithm with simple example, two time series can be taken into consideration. Let,

$$\mathbf{A} = [1, 2, 5, 8, 9, 2, 1, 5, 7, 3]$$

$$\mathbf{B} = [1, 5, 3, 4, 1, 9, 4, 3, 6, 3]$$

The time series A is plotted in the vertical axis and the time series B is plotted in the horizontal axis. The first step in the algorithm is to take a matrix of size  $m \times n$ ,

Where,  $m$  = no. of values in A

$n$  = no. of values in B.

After taking the matrix, all the indices are assigned a higher value, hypothetically infinity. The pseudocode of the first step is as follows:

```

for i = 1 to m
  for j = 1 to n
    DTW [ i, j] = infinity;

```

It becomes easier to compute the values of the first row and column if the initial value is set to a larger value. As we iterate through the whole matrix by two for loops, the calculation for each index follows the pseudocode below:

$$|A_i - B_j| + \min \begin{cases} D [ i-1, j-1], \\ D [ i-1, j], \\ D [ i, j-1] \end{cases}$$

For the chosen two time series, calculation of the DTW matrix is depicted as below:

Table 3: Path calculation of time series A and B using DTW

3	33	19	17	16	17	23	18	17	17	14
7	31	17	17	15	18	17	17	18	14	17
5	25	15	13	12	15	15	14	14	13	15
1	21	15	11	12	11	19	13	12	16	17
2	21	11	9	11	12	15	10	11	15	16
9	20	8	10	11	14	8	13	18	19	22
8	12	4	6	6	10	8	12	17	16	20
5	5	1	2	3	7	11	12	14	15	17
2	1	3	4	6	7	14	16	17	21	22
1	0	4	6	9	9	17	20	22	27	29
	1	5	3	4	1	9	4	3	6	3

The path pairs of the two analysed time series calculated via DTW algorithm are as follows:

$[(0, 0), (1, 0), (2, 1), (2, 2), (2, 3), (2, 4), (3, 5), (4, 5), (5, 6), (6, 7), (7, 8), (8, 8), (9, 9)]$

After calculating all the indices, the values are selected from the top of the matrix. Again, the minimum among the lower right, lower left and diagonally left values are taken into consideration to decide which value to choose. The DTW distance for the given two time series are shown in green colour in the matrix. It can be seen from the matrix that we are able to get same values multiple times which avoids the point-to-point comparison. Thus, the whole signal is taken into consideration.

The DTW distance calculation of the above algorithm by analysing two time series A and B can be shown as below:

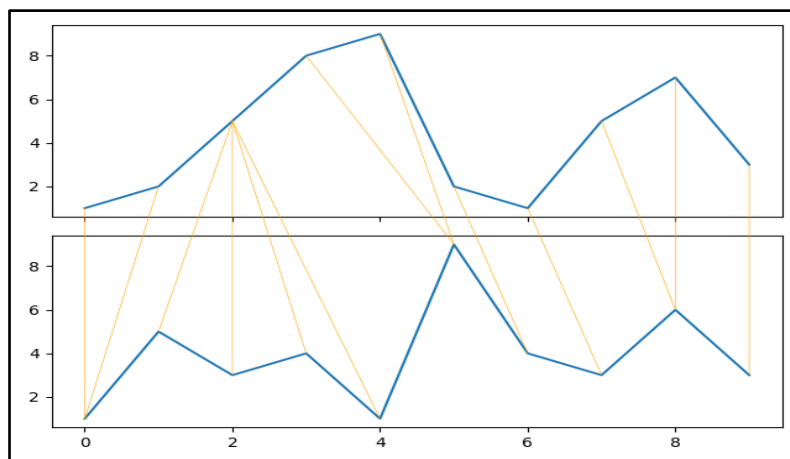


Figure 15: Distance calculation of time series A and B using DTW

## **DTW Time Complexity**

Based on the pseudocode above, the time complexity of the DTW algorithm is  $O(NM)$ , where  $N$  and  $M$  are the lengths of the two input sequences. Both sequences are compared with linear complexity of  $O(N + M)$ .

## **DTW Weakness/Disadvantages**

The DTW has some weaknesses. First,  $O(n^2v)$  complexity may not be sufficient for a larger vocabulary, which could lead to an increase in the recognition rate. Additionally, evaluating two parts from two different sequences is difficult because of the various channels with various properties.

## **Project Focus DTW implementation**

Dynamic programming is simple to implement, and DTW is a good demonstration of dynamic programming. While on its own, DTW has average speaker-independent performance. There needs to be practical training examples for the comparison to be made. When it comes to continual recognition activities, it is prone to failure. Despite having developed a straightforward voice recognition system utilizing DTW, we included a proof of concept by building a system that incorporated DTW along with other features such as word detection and MFCC features standard deviation for testing purpose. Limited by time and resource constraints, the project wasn't able to extend the library of training voice as this would require extensive level of investigation and calculations.

After executing the programme and determining the appropriate emotion detection threshold values. We discovered that any voice with an MFCC value distance comparison of less than 50 had the correct sentiment. As a result, we have configured the algorithm to continue the loop process of identifying the correct emotions only if the DTW distance is less than 50 when compared to the lowest distance identified between the training and input voices. The code component is shown below figure 16.

```

#loop through the distance[]
for i in range(len(distance)):
    #if distance is less than 50 then continue with operation else exit the loop
    if mindist < 50:
        #identify the position of the minimum distance
        if mindist == distance[i]:
            print("The minimum distance matched is", distance[i])
            print("The emotion is:", emotion[i])
            mfccs1 = mfccsf[i]
            #write the time warping graph into the file by comparing the distance of the min distance identified.
            path = dtw.warping_path(mfccs, mfccs1)
            dtwvis.plot_warping(mfccs1, mfccs, path, filename="warp.png")
            #call the right animoji based on the min distance found above - opening another py file
            os.system(file[i])
        elif mindist > 50:
            break
    if mindist > 50:
        #output unidentified animoji if distance is above 50
        os.system('python unidentified.py')

```

Figure 16: Threshold value set for emotion detection

## Testing and Quality Checks

We have generated test cases and test scenarios based on the program that we developed in order to identify which technique gives the most accurate result. Hence based on the figure below we can see there is 3 test cases and 4 test scenarios used to test the program and we have also included positive and negative test within the scope. From the test cases and scenarios, we have generated the result in a graph form for better visualisation of the test result.

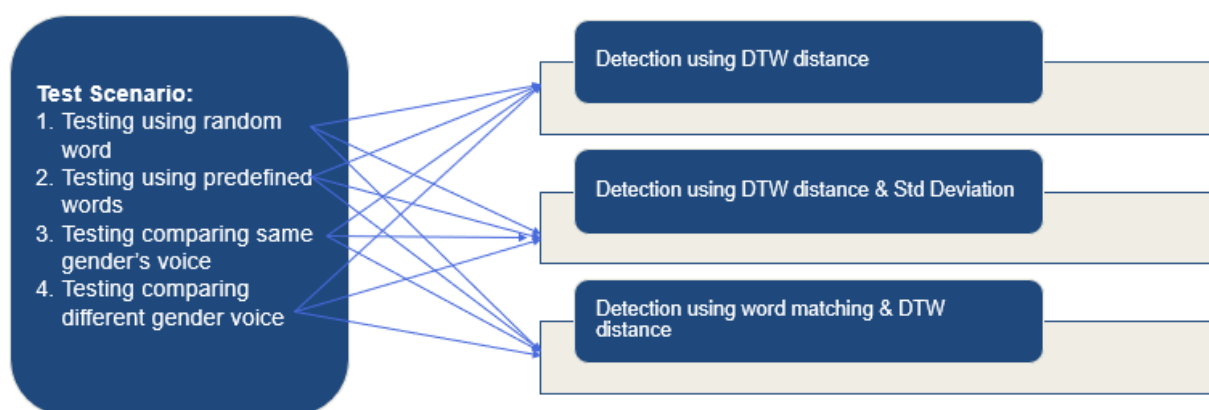


Figure 17: Test Cases and Test Scenario

## Test Result

Testing was conducted in accordance with the procedure outlined in the following figure. Although the procedure is straightforward, the total process involves extensive test quality validation by all team members.

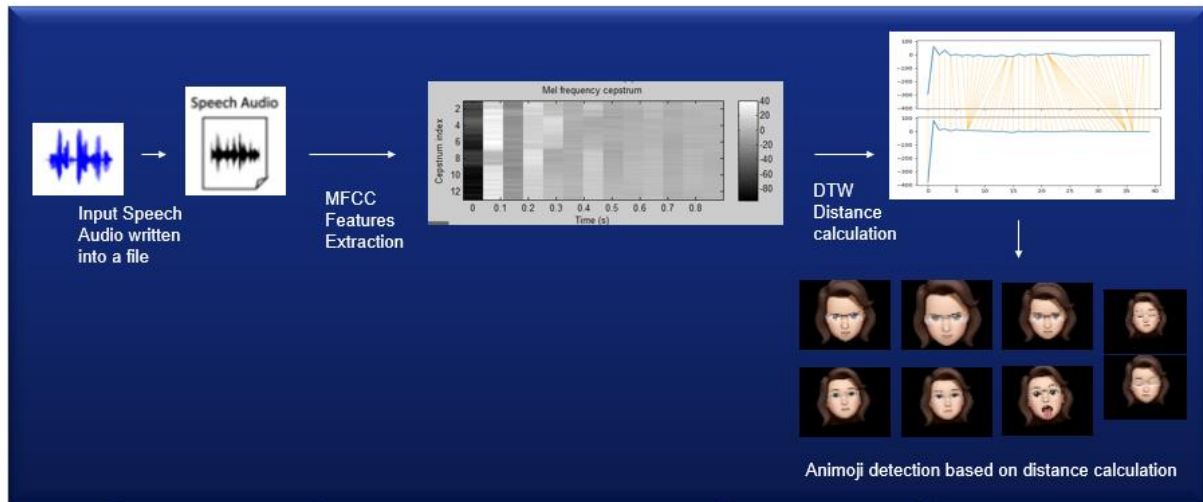


Figure 18: Testing process flow

Below results are based on 60 tries done 20 times by each team members in order to establish reliable result. The total is taken and averaged to get the overall result that is shown in the second part of the result displayed in a graphical overview.

SPECIFIC WORDS TEST							
Female to female Voice Comparison							
Test Case	Anger1	Anger2	Anger3	Disgust1	Disgust2	Disgust3	Neutral
DTW Lowest Distance	16	17	10	17	17	17	18
DTW Lowest Distance & Std Deviation	11	10	7	8	8	9	9
DTW distance and word detection	11	9	6	13	12	7	18
Male to male Voice Comparison							
Test Case	Anger1	Anger2	Anger3	Disgust1	Disgust2	Disgust3	Neutral
DTW Lowest Distance	20	18	17	10	15	17	13
DTW Lowest Distance & Std Deviation	4	8	16	0	2	16	1
DTW distance and word detection	18	19	18	19	19	18	18
Male to female Voice Comparison							
Test Case	Anger1	Anger2	Anger3	Disgust1	Disgust2	Disgust3	Neutral
DTW Lowest Distance	6	3	6	2	5	6	12
DTW Lowest Distance & Std Deviation	4	3	6	2	3	3	9
DTW distance and word detection	0	0	0	0	0	0	0

Figure 19: Test Result using specific words

## Test Cases and Result With Specific Words

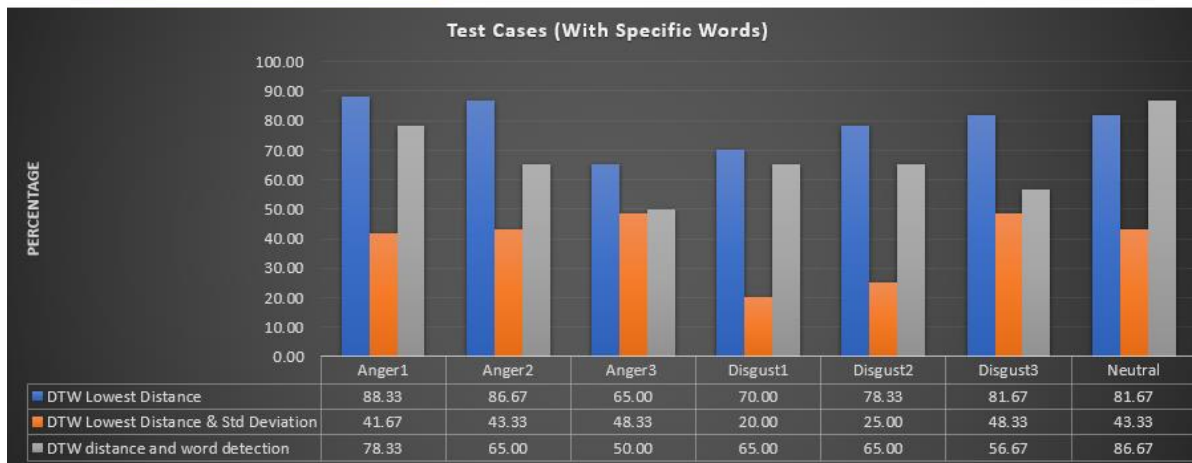


Figure 20: Average Test Result with Specific Result

## RANDOM WORDS TEST

### Female to female Voice Comparison

Test Case	Anger1	Anger2	Anger3	Disgust1	Disgust2	Disgust3	Neutral
DTW Lowest Distance	18	16	12	18	17	17	15
DTW Lowest Distance & Std Deviation	8	6	4	4	6	7	8
DTW distance, Word detection and Standard Deviation	0	0	0	0	0	0	0

### Male to male Voice Comparison

Test Case	Anger1	Anger2	Anger3	Disgust1	Disgust2	Disgust3	Neutral
DTW Lowest Distance	4	9	14	1	5	5	4
DTW Lowest Distance & Std Deviation	1	4	1	0	4	3	1
DTW distance and word detection	0	0	0	0	0	0	0

### Male to female Voice Comparison

Test Case	Anger1	Anger2	Anger3	Disgust1	Disgust2	Disgust3	Neutral
DTW Lowest Distance	13	14	11	16	14	7	12
DTW Lowest Distance & Std Deviation	12	9	7	3	5	6	9
DTW distance and word detection	0	0	0	0	0	0	0

Figure 21: Test result using random words

## Test Cases and Result With Random Words

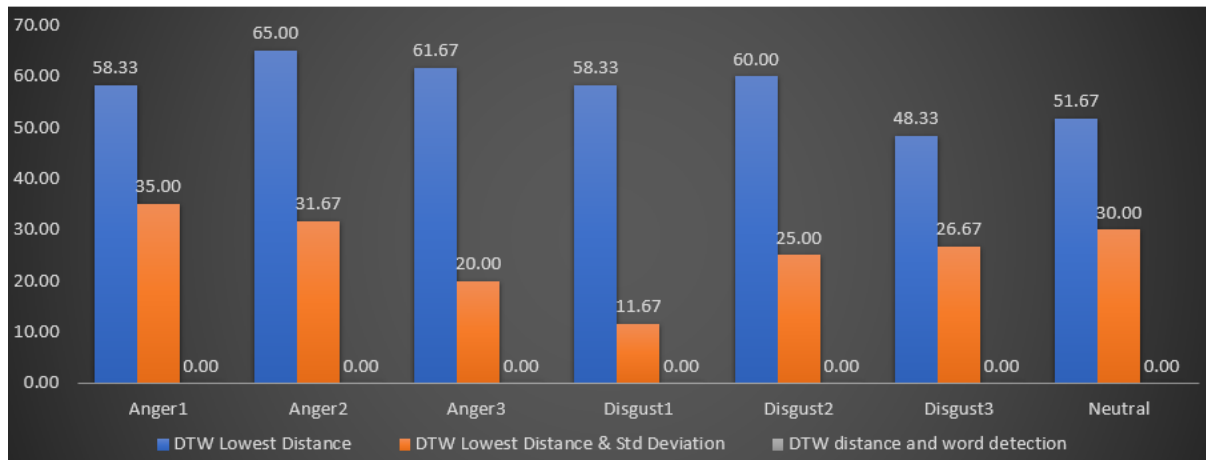


Figure 22: Average Test Result with Specific Result

### Positive test

In this section the testing process, outcome and results will be discussed. The testing in week 2 was done to detect the emotion based on the time warping distance and the words used in the input voice as compared to the training voice. In this example, input voice is compared to a single output voice due to the time and resources constraint. The image in figure 1, 2 and 3 below shows the example of the success result whereby the time warping distance between the two voice was detected to be as of 29 and the use of word 'Angry' in the sentence. The training voice was recorded by saying 'I'm angry' while the input voice was saying 'I'm angry I don't know what to say'.

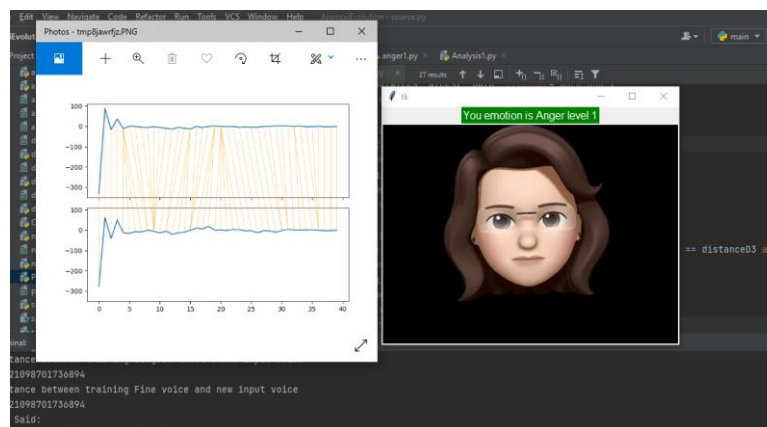


Figure 23: Anger Level 1 female voices only

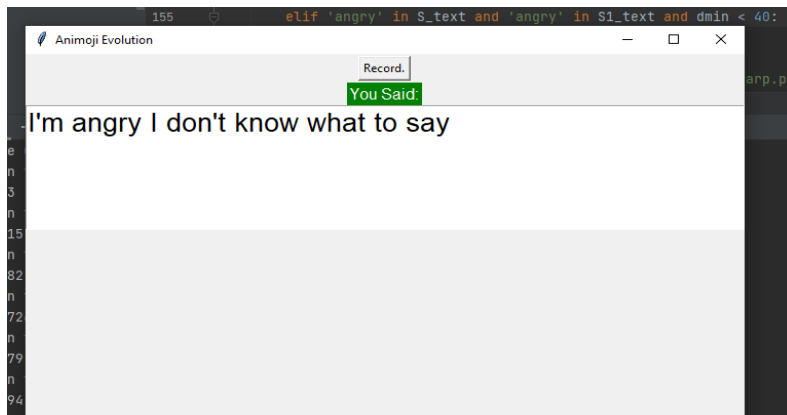


Figure 24: Anger level 1 input text comparison

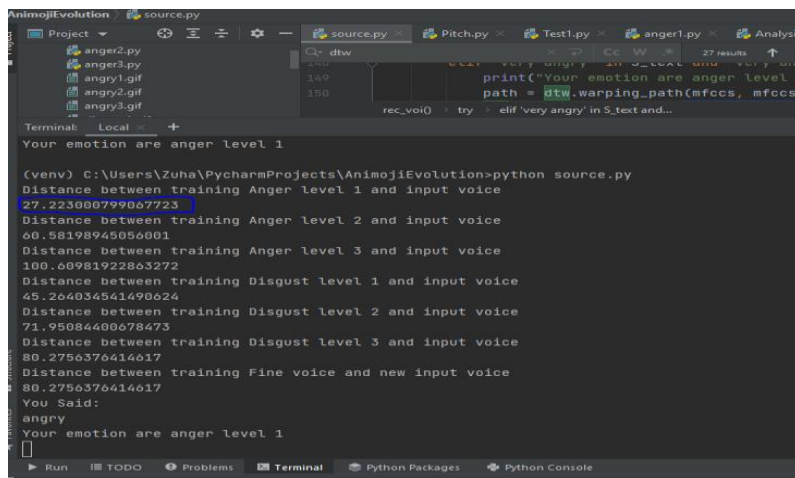


Figure 25: Anger level 1 :time warping distance

In the next test case shown in figure 4, 5, 6 and 7 below are the testing done with a different person voice to compare male and female voice to see if the voices can be matched. The training voice is recorded with female voice while testing was done by male voice, and it was matched after several try to match the tone and pitch used in the training voice.

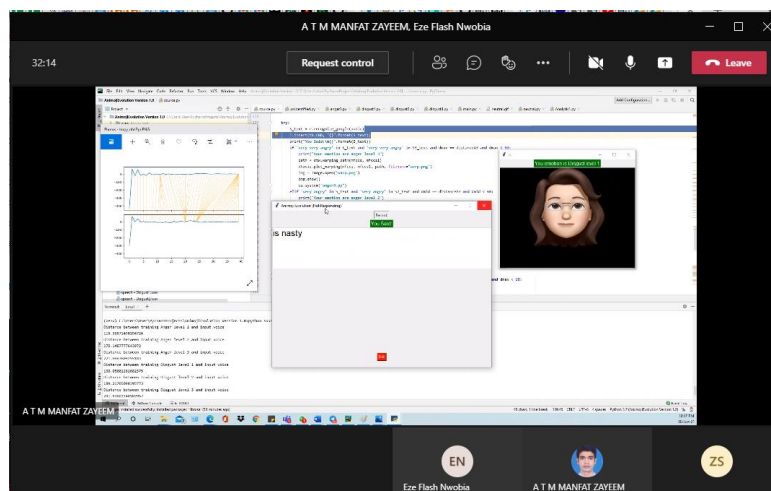




Figure 26: Disgust Level 1 male and female voice comparison

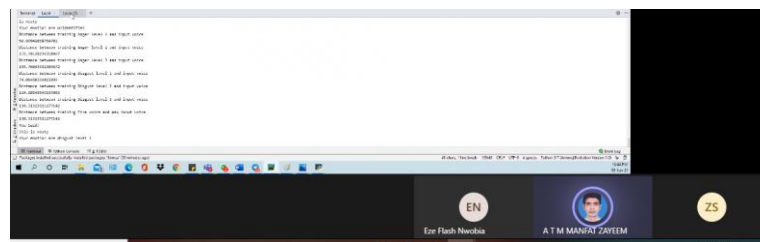


Figure 27: Disgust level 1-time warping result

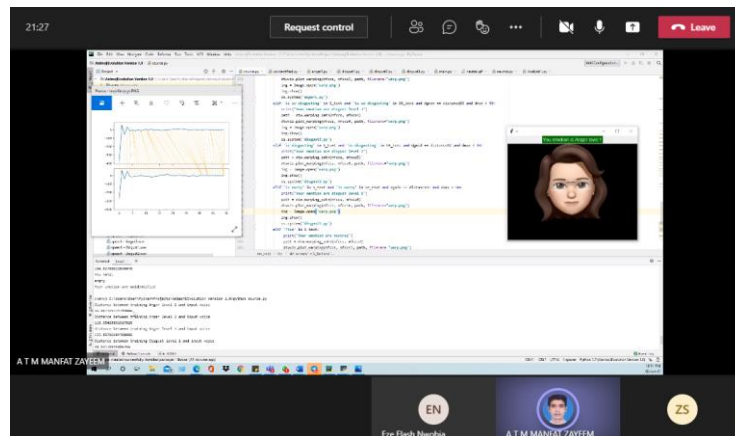


Figure 28: Anger level 1 male and female voice comparison

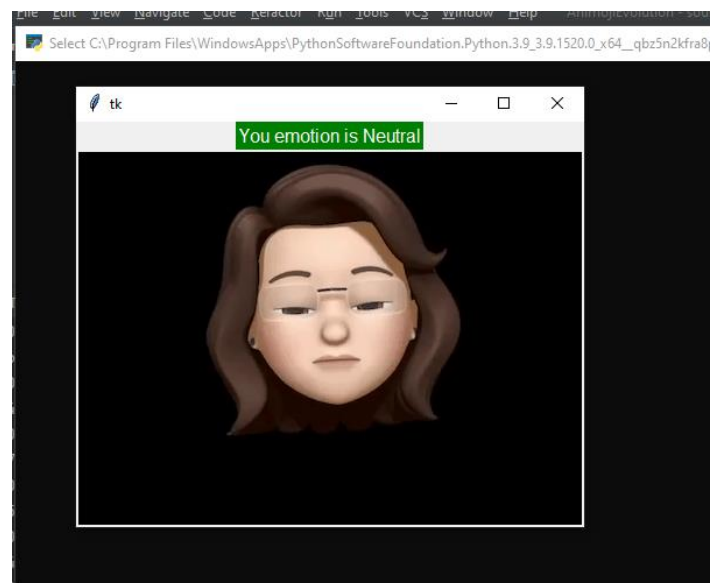


Figure 29: Neutral feeling success

```
122.2751332027685
Distance 2
170.66360854282098
Distance 3
57.461724823642726
Distance 4
126.35801172050921
Distance 5
138.87007641118007
Distance 6
35.21952883476018
35.21952883476018
You Said:
just thinking
The minimum distance is 35.21952883476018
The minimum standard deviation is 1.6359711
The emotion is: Neutral
The standard deviation value is: 1.6359711
```

```
Terminal: Local x +
44.16831477330455
Distance between training Anger level 2 and input voice
91.96351880383925
Distance between training Anger level 3 and input voice
138.3251360982069
Distance between training Disgust level 1 and input voice
38.50284511836645
Distance between training Disgust level 2 and input voice
97.22972738170135
Distance between training Disgust level 3 and input voice
108.89932770388202
Distance between training Fine voice and new input voice
108.89932770388202
You Said:
I'm feeling fine
Your emotion are neutral
```

Figure 30: Neutral feeling output – time warping distance

## Negative Test

In this section the example results of negative test will be presented with few screenshots. The negative test is done by comparing wrong use of words, wrong pitch and tone as well as by using male and female voice comparison. Figure 9 and 10 shows failed comparison of voices for anger level 1 while using male and female voice comparison.

In figure 11 and 12 below shows the failed test to get disgust level 2 emotion when comparing male and female voice. The result comes out as unidentified emotion due to mismatch of the comparison values.

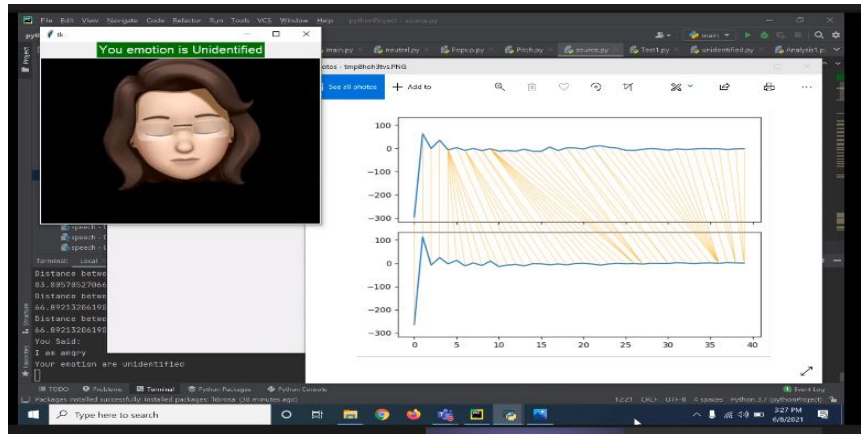


Figure 31: Fail testing for anger level 1 male vs female voice

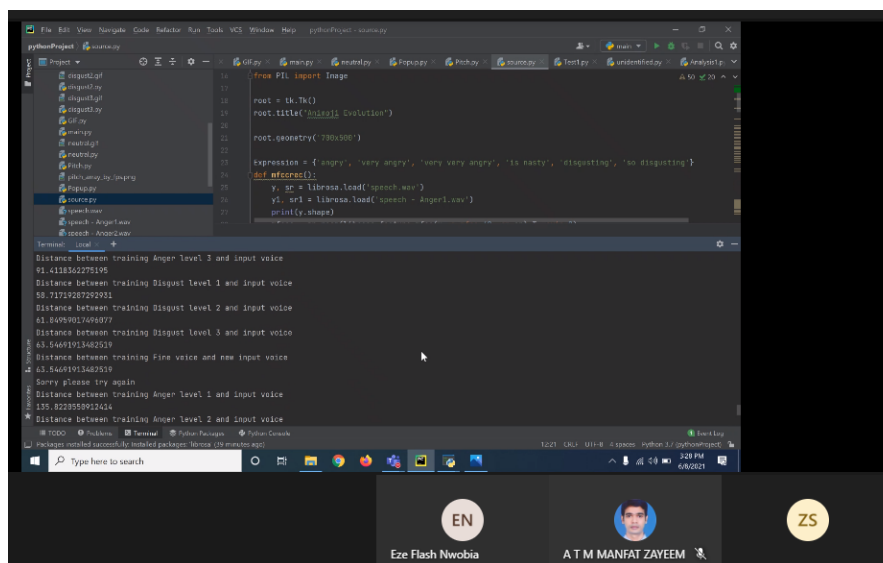


Figure 32: Anger level 1 fail male vs female voice comparison

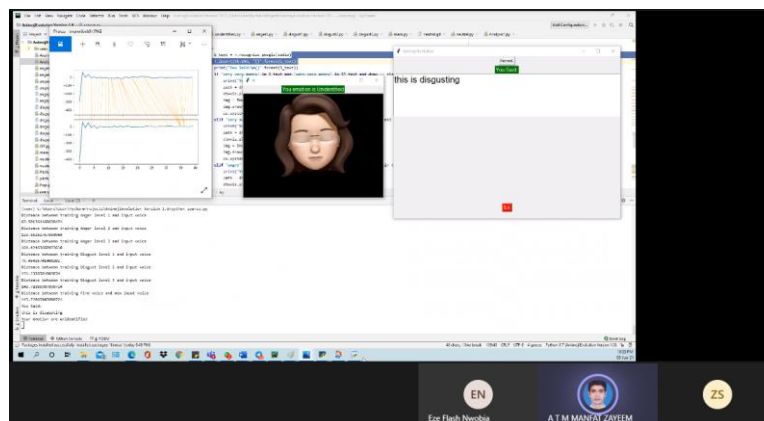


Figure 33: Disgust level 2 fails male vs female voice comparison

```
Distance between training Anger level 2 and input voice
123.65262767854944
Distance between training Anger level 3 and input voice
164.42463660933618
Distance between training Disgust level 1 and input voice
79.45485741400202
Distance between training Disgust level 2 and input voice
131.1326504003834
Distance between training Disgust level 3 and input voice
143.72865907890724
Distance between training Fine voice and new input voice
143.72865907890724
You Said:
this is disgusting
Your emotion are unidentified
```

Figure 34: Disgust level 2 fail with time warping value

#### DS4: Create an interface or GUI for the above

The program built for the Animoji emotion detection is a desktop based and we have created a user-friendly graphical user interface as per the figures below. The image below is self-explanatory since it is using a very simple GUI and basic user functions.

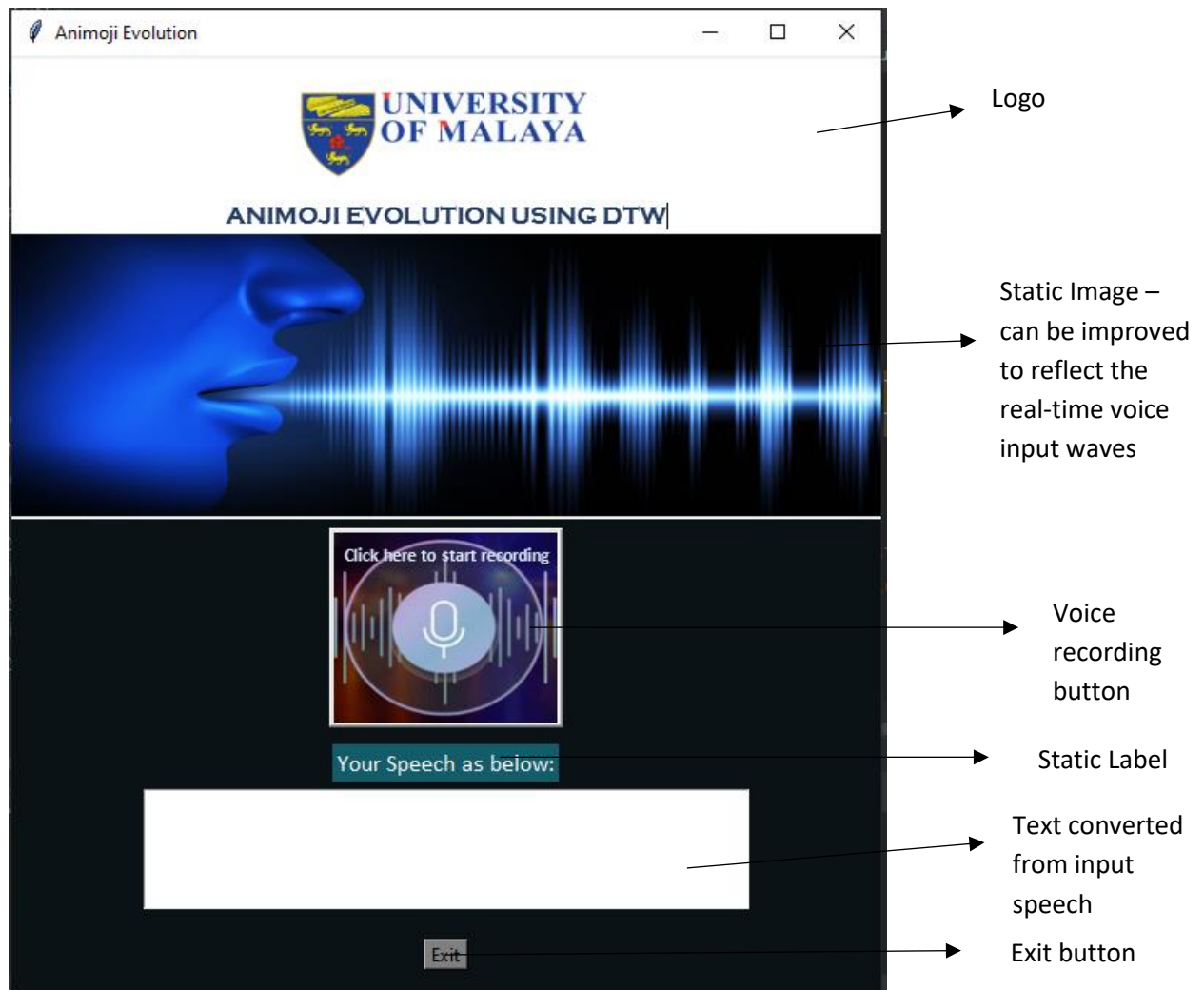


Figure 35: Main Program GUI

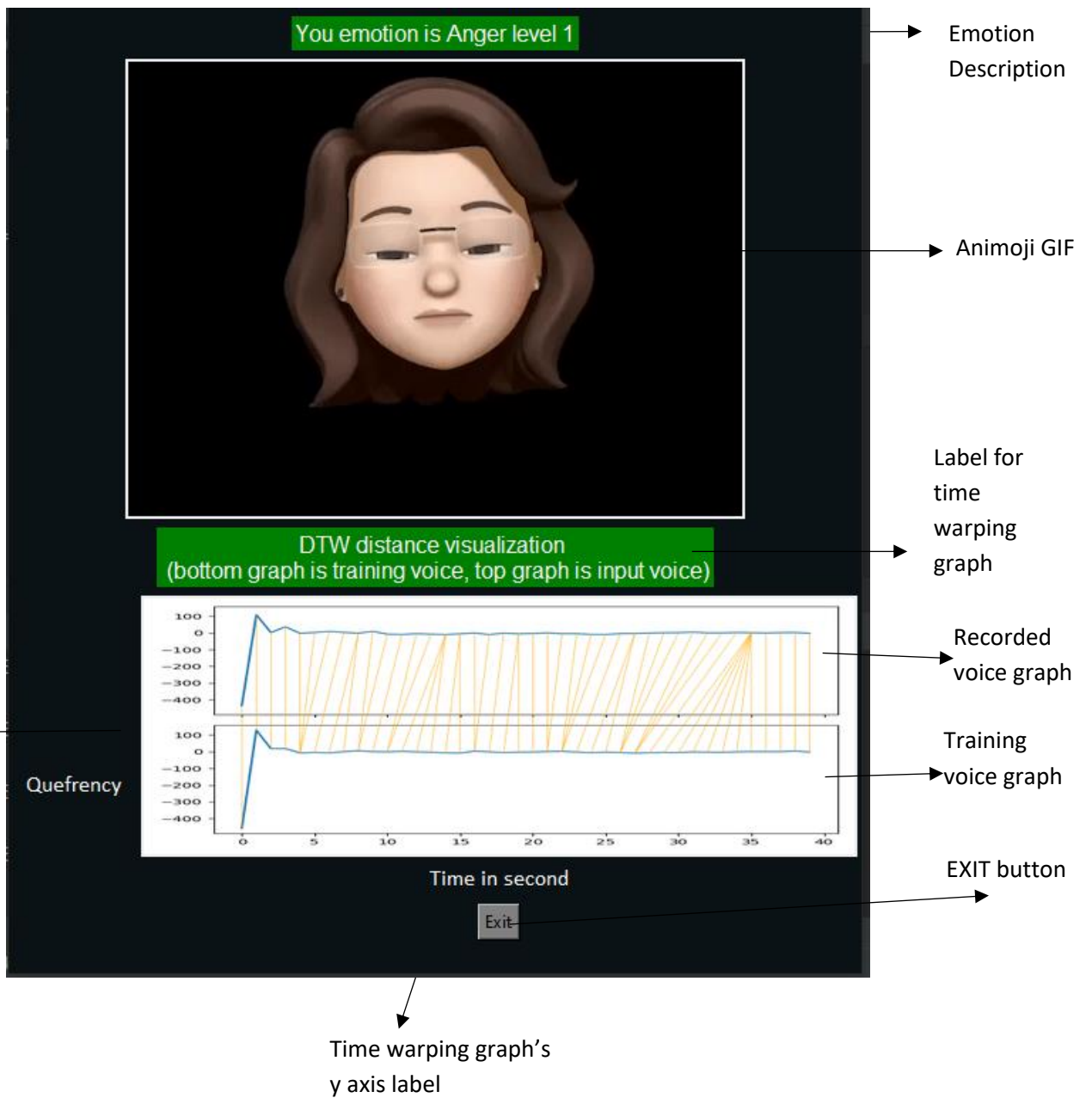


Figure 36: Program's Output GUI

# Additional Output File

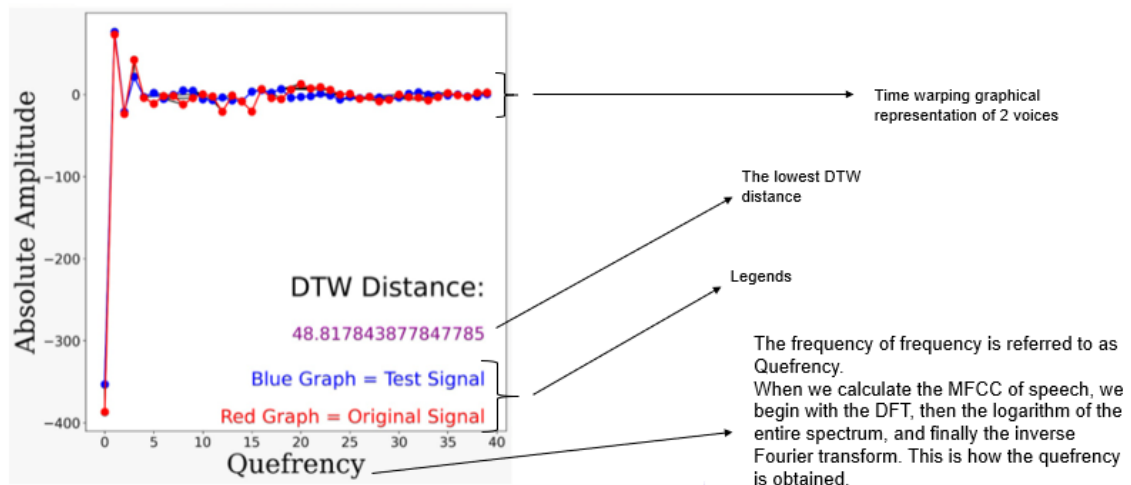


Figure 36a: Program's Output GUI

Aside from the program GUI, our team has also developed a website to promote the project and program done as part of the project. In the website we can get an overview of the project from the administration level to the technical details.

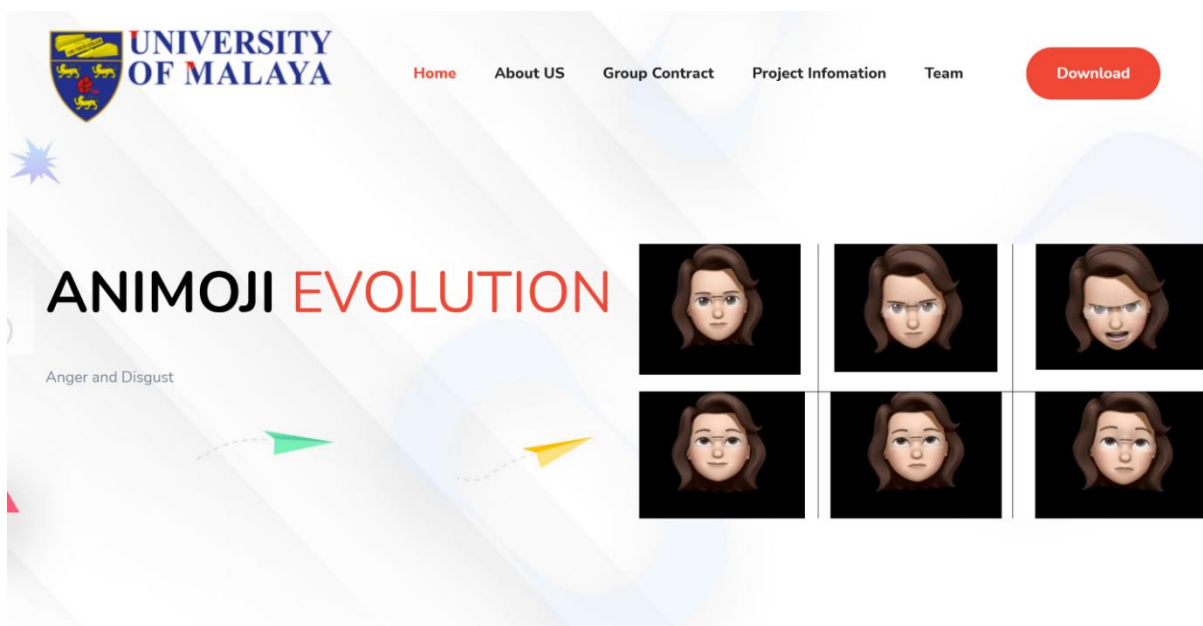


Figure 37: Homepage of Animoji Evolution



## FLASH

[illegible]

Figure 39: Project Information



## **Overall reflection of the experience doing the project**

Our group Animoji has chosen to implement anger and disgust emotion recognition as part of the project's requirements and scope. Through this Animoji project, we receive exposure to complex techniques such as Dynamic Time Warping and Python, as well as to rapid project development. Our group was given four weeks to complete our works, which is considered a rapid development cycle. We met weekly to review progress and provide input on any concerns that occurred. DTW is excellent for voice recognition as it can adapt to various speech rates. Each week, Dr. Raja Jamilah interviewed us about our progress; by attending such sessions, we were able to pick up new skills and swiftly fill up the gaps identified by Dr Raja Jamilah based on her vast experience in this field. We feel that this procedure, guided by Dr. Raja Jamilah, has increased our knowledge and practical experience. We appreciate the collaboration and efforts of our group members in technical areas such as coding, testing, integration, and accuracy enhancement. Despite the limits of the project, we group gave it our all.

## Conclusions

Apple released Animoji, or animated emoji, on electronic devices in 2017. Emojis that are animated are referred to as Animojis. Our project was comparable to many others that detect emotions in voice, however our need required us to apply the Dynamic Time Warp (DTW) algorithm to distinguish emotions in terms of frequency and intonation in audio or spoken. In general, this Animoji project's planning, research, and implementation went smoothly. Even though we encountered difficulties in completing the project, we were able to handle them rather well by using an additional method such as MFCC.

Our project's primary strengths were its exploration and investigation on the internet, as well as its search for answers and references to address our bugs and jobs, as our aim was thoroughly researched. Another critical strength is the spirit, and we worked tirelessly until the last minute to integrate the system's back-end. While our project's shortcoming was a lack of Python programming skills among team members, we were nevertheless able to learn and develop the software in a short period of time through extensive discussion and communication. As this was a collaborative effort, each participant offered their own strengths and participated in both the brainstorming and technical development processes. Finally, teamwork is a critical component of effective work, and we believe we have attained it, aided by our lecturer, Dr. Raja Jamilah.

## Appendix

Steps to run the developed program.

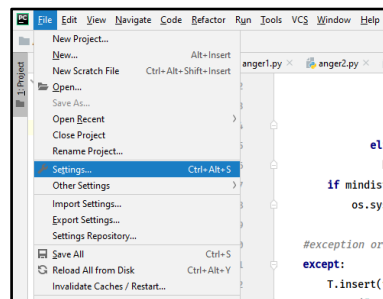
### 10. Installing PyCharm python IDE.

- Before installing PyCharm, Python should be installed in the machine. Particularly, for this project Python 3.7.9 (64 bit) was used. The link is as below:  
<https://www.python.org/downloads/release/python-379/>
- Downloading PyCharm from the JetBrains website. The link is below:  
<https://www.jetbrains.com/pycharm/download/>  
(Community Version was downloaded as it is free and opensource)
- Creating a new Python project after opening PyCharm IDE.
- After Creation of the project, some required modules should be installed from the PyCharm IDE and some modules should be installed from the terminal integrated in the PyCharm IDE. The next section describes step by step on how to install those modules / libraries.

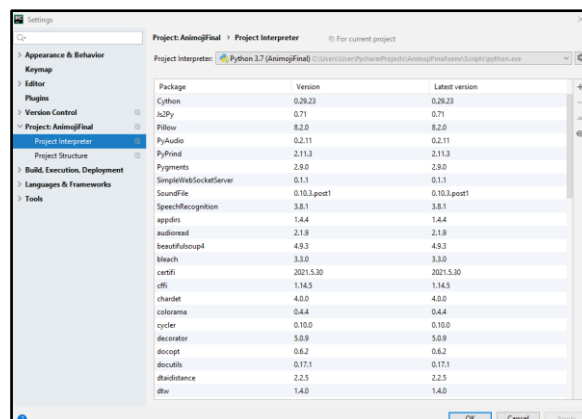
### 11. Installing required libraries / modules from PyCharm IDE.

The required libraries can be installed in PyCharm by going to:

File -> Settings



In the settings, under the project name, the 'Project Interpreter' tab will show all the packages / libraries installed in that specific project.



For this Animoji Evolution Project the below libraries were installed while developing:

- 1) **Matplotlib** – for plotting graphs
- 2) **Numpy**
- 3) **Pillow** – for accessing Images.
- 4) **DTW** – for calculating the DTW distance of two speech signals.
- 5) **Librosa** – for retrieving the MFCC features of the speeches.

The above-mentioned modules were installed directly from the IDE (Project Interpreter). But there were some modules that needed to be installed from the IDE's terminal. They are as follows:

- 1) SpeechRecognition – To convert speech to text.  
Terminal Command: pip install SpeechRecognition
- 2) PyAudio.

For installing PyAudio the below steps were followed:

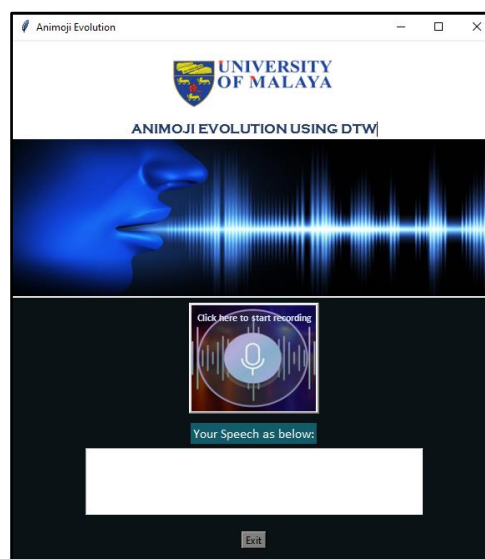
- i. Installing pipwin:  
Terminal Command: pip install pipwin  
After installing pipwin, the below command was given:  
Terminal Command: pipwin install pyaudio

## 2. Running the program

To run the developed python program, from the terminal the below command should be entered:

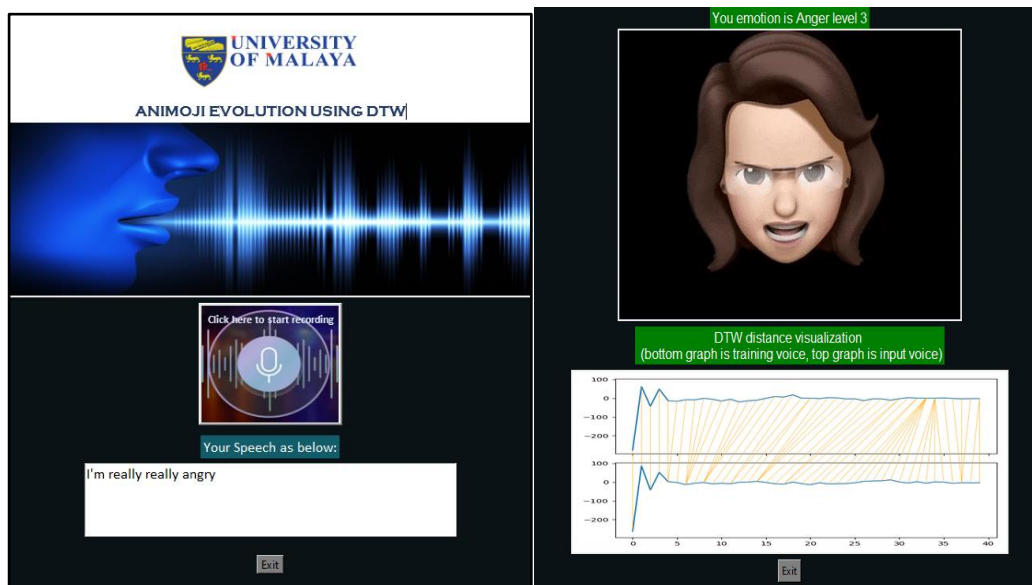
Terminal Command: Python source1.py

The above command will pop up the below window which is the main GUI of the program.



By clicking the 'microphone' button, the program will start to receive sounds from the microphone installed in the machine. After receiving the speech the program will analyse the MFCC features of the speech and will try to match the speech according to the algorithm developed at the back of the program. In the backend, some speech's MFCC values are saved. The program to try to match and will try to predict whether it is an angry speech or disgust speech. If the sentence does not fall into these two categories, the program will check some values of the speech whether to pop up neutral Animoji. If the above category also fails, it will fall into unidentified Animoji.

A sample case:



**Figure: Program Detecting the speech**

**Figure: Program popping up Animoji**

## References

- Umamaheswari, J., & Akila, A. (2019, February). An enhanced human speech emotion recognition using hybrid of PRNN and KNN. In *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)* (pp. 177-183). IEEE.
- Kexin, T., Yongming, H., Guobao, Z., & Lin, Z. (2019, November). Research on Emergency Parking Instruction Recognition Based on Speech Recognition and Speech Emotion Recognition. In *2019 Chinese Automation Congress (CAC)* (pp. 2933-2937). IEEE.
- Cherry, K. (2020, January 13). *The 6 Types of Basic Emotions and Their Effect on Human Behavior*. verywellmind. <https://www.verywellmind.com/an-overview-of-the-types-of-emotions-4163976>
- El-Yamri, M., Romero-Hernandez, A., Gonzalez-Riojo, M., & Manero, B. (2019). Designing a VR game for public speaking based on speakers features: a case study. *Smart Learning Environments*, 6(1), 1-15.
- Cowen, A. S., & Keltner, D. (2017). Self-report captures 27 distinct categories of emotion bridged by continuous gradients. *Proceedings of the National Academy of Sciences*, 114(38), E7900- E7909.
- Koops, T. "One More Thing..."—A critical approach to the Apple 2017 Keynote Presentation.
- Blagdon, J. (2013). How emoji conquered the world. *The Verge*, 4.
- Evans, V. (2017, August 17). *Emojis actually make our language better*. nypost. <https://nypost.com/2017/08/12/emojis-actually-make-our-language-way-better/>
- Likitha, M. S., Gupta, S. R. R., Hasitha, K., & Raju, A. U. (2017, March). Speech based human emotion recognition using MFCC. In *2017 international conference on wireless communications, signal processing and networking (WiSPNET)* (pp. 2257-2260). IEEE.
- XinXing, J., Xu, S. (2012). Speech Recognition Based on Efficient DTW Algorithm and Its DSP Implementation. In *International Workshop on Information and Electronics Engineering (IWIEE)*. Elsevier.
- Simon, D., Becker, M., Mothes-Lasch, M., Miltner, W.H.R., Starube, T. (2016). Loud and angry: sound intensity modulates amygdala activation to angry voices in social anxiety disorder. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5390751/>. NCBI.
- Chen, X., Yang, J., Gan, S., Yang, Y. (2012). The Contribution of Sound Intensity in Vocal Emotion Perception: Behavioral and Electrophysiological Evidence. <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3264585/>. NCBI.

Zhang, J. (2020, February 1). *Dynamic Time Warping*. towardsdatascience.

<https://towardsdatascience.com/dynamic-time-warping-3933f25fcdd>

Seunagal, G. (2021, February 15). *Are there Different Levels Of Anger?*. betterhelp.

[https://www.betterhelp.com/advice/anger/are-there-different-levels-of-anger/?utm\\_source=AdWords&utm\\_medium=Search\\_PPC\\_c&utm\\_term=\\_b&utm\\_content=118051370527&network=g&placement=&target=&matchtype=b&utm\\_campaign=11771068538&ad\\_type=text&adposition=&gclid=CjwKCAjwiLGGBhAqEiwAgq3q\\_szCcI\\_iGtMHksilhq3x6X-Tmsck2qLq5MRKE6vBwT9lRUjGZOyTxoCB64QAvD\\_BwE](https://www.betterhelp.com/advice/anger/are-there-different-levels-of-anger/?utm_source=AdWords&utm_medium=Search_PPC_c&utm_term=_b&utm_content=118051370527&network=g&placement=&target=&matchtype=b&utm_campaign=11771068538&ad_type=text&adposition=&gclid=CjwKCAjwiLGGBhAqEiwAgq3q_szCcI_iGtMHksilhq3x6X-Tmsck2qLq5MRKE6vBwT9lRUjGZOyTxoCB64QAvD_BwE)

Fong, S. (2012). Using hierarchical time series clustering algorithm and wavelet classifier for biometric voice classification. *Journal of Biomedicine and Biotechnology*, 2012.

Ekman, P. (2021). *Disgust*. Paulekman. <https://www.paulekman.com/universal-emotions/what-is-disgust>

Velardo, V. [ Valerio Velardo - The Sound of AI]. (2020, October 5). *Mel- Frequency Cepstral Coefficients Explained Easily* [Video]. YouTube.

[https://www.youtube.com/watch?v=4\\_SH2nfbQZ8&t=960s&ab\\_channel=ValerioVelardo-TheSoundofAI](https://www.youtube.com/watch?v=4_SH2nfbQZ8&t=960s&ab_channel=ValerioVelardo-TheSoundofAI)

Wolfe, J. (n.d.). *Voice Acoustics: an Introduction*. newt.phys.unsw. h  
<https://newt.phys.unsw.edu.au/jw/voice.html>