Offensive Language Detection Using Machine Learning (OLDUM)

By

Ijaz Shah

Hamza Khalid

Muhammad Hasnat Nasir

Zeeshan Karimi

Mubashir Arman

Supervised by:

Prof Dr. Shibli Nisar

Co Supervised by:

Prof Dr. Alina Mirza

Submitted to the faculty of Department of Electrical Engineering,
Military College of Signals, National University of Sciences and Technology, Islamabad,
in partial fulfillment for the requirements of B.E Degree in Electrical (Telecom) Engineering.

May 2022

In the name of ALLAH, the Most Merciful, the Most Benevolent

CERTIFICATE OF CORRECTNESS AND APPROVAL

This is to officially state that the thesis work contained in this report

"Offensive Language Detection Using Machine Learning"

is carried out by

Ijaz Shah

Hamza Khalid

Muhammad Hasnat Nasir

Zeeshan Karimi

Mubashir Arman

under my supervision and that in my judgement, it is fully ample, in scope and excellence, for the degree of Bachelor of Electrical (Telecom.) Engineering in Military College of Signals, National University of Sciences and Technology (NUST), Islamabad.

Approved by

Supervisor

Prof Dr. Shibli Nisar

Department of EE, MCS

Date: 26-5-2022

DECLARATION OF ORIGINALITY

We hereby declare that no portion of work presented in this thesis has been submitted in support of another award or qualification in either this institute or anywhere else.

ACKNOWLEDGEMENTS

Allah Subhan'Wa'Tala is the sole guidance in all domains.

Our parents, colleagues and most of all supervisor, **Prof Dr. Shibli Nisar** without your guidance.

The group members, who through all adversities worked steadfastly.

Plagiarism Certificate (Turnitin Report)

This thesis has __10__ similarity index. Turnitin report endorsed by Supervisor is attached.

IJAZ SHAH
00000242844

HAMZA KHALID
00000280707

MUHAMMAD HASNAT NASIR
00000278787

ZEESHAN KARIMI
00000280711

MUBASHIR ARMAN (CSE)
00000201374

Asst. Prof Shibli Nisar

(Supervisor)

ABSTRACT

Cyberbullying using offensive language on the Internet has become a major problem among all age groups. Automatic detection of offensive language from social media applications, websites, and blogs is a difficult but important task. In recent years, the presence of offensive language on social media platforms and automatic detection of such language is becoming a major challenge in modern society. The complexity of natural language constructs makes this task even more challenging. Until now, most of the research has focused on resource-rich languages like English. This study is about the detection of offensive language from the user's audio presented in a resource-poor language i.e., Pushto. We propose the first offensive dataset of Pushto containing user-generated Audio from social media. We use individual and combined n-grams techniques to extract features at word level and gender basis. We will apply classifiers from different machine learning techniques to detect offensive language from Pushto Audio.

Offensive Language detection Using Machine Learning (OLDUM) aims at developing a prototype of a system that, using machine learning, will be capable of detecting offensive words in Pashto language, helping in automating the process of AUDIO/VOICE note by the social media Applications/Website and therefore stopping any unethical activity.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

OLDUM - Offensive Language Detection Using Machine Learning

ASR – Automatic Speech Recognition

ADC – Analog to Digital Converter

MFCC – Mel-Frequency Cepstrum Coefficient

UI – User Interface

ML – Machine Learning

CNN- Convolutional Neural Network

SVM - A support vector machine

RNN - Recurrent neural networks

MLP - A Multilayer perceptron

Chapter 1: Introduction

Offensive Language Detection Using Machine Learning can be used to detect offensive words in Pushto Language, these words can be taken from a large number of audios and videos. Monitoring this large amount of audios manually is a vert difficult task, OLDUM does all this automactically monitoring the use of offensive words by social media platforms, therefore making it possible to stop any offensive activity.

1.1 Problem Definition

Monitoring of a huge amount of data by the social media platforms manually is a very difficult if not impossible job and this difficulty increases with the increase in the use of these social media platforms [1]. Monitoring everything manually is a serious problem faced by social media applications i.e., WhatsApp, Facebook, YouTube etc. researchers have made a lot of efforts to automate this process using ML. There are a lot of systems for Urdu and English and all the other rich languages, but till now there has not been a lot of work done in this domain in Pushto.

1.2 Proposed Solution

OLDUM aims at using machine learning to make a system which is capable of detecting offensive and vulgar words in Pashto language, helping in automating the process of audio tracing by social media platforms and security agencies and thus stopping use of offensive words.

1.3 Scope of Project

Till now our system can only process single words at a time, the model has only been trained to detect pre decided words in Pushto languages. Initially, OLDUM will be trained for isolated words only. In the near future the system can be changed to detect spontaneous speech.

1.4 Objectives

OLDUM aims at tracking offensive words automatically in Pushto. During this project, all the aspects of software engineering are covered i.e., survey and feasibility analysis, requirement gathering, architectural and detailed design, implementation and testing along with documentation.

1.5 Deliverables

Sr No.	**Tasks**	**Deliverables**
1	Literature Review	Review
2	Requirements Specification	Hardware (Raspberry pi)
3	Detailed Design	Software Design Specification
4	Implementation	Demonstration of system
5	Training	System deployment
6	Testing	System evaluation
7	Deployment	Necessary documentation

1.6 Justification for Selection of Topic

With increasing advancement in social media and communication, an increase in offensive content has been spotted. Even though Pashto is spoken by a huge population, Automatic Speech Recognition for the Pashto is still not being explored to an extent as compared to other languages [2].

1.7 Document Overview

The complete working of the OLDUM is shown in this document. Starting with literature review i.e. the requirement of the system, architecture of the system, modules of the software, and the system in form of component diagram.

1.8 Structure of Thesis

These comprises of following chapters:

Chapter 2: Problem statement and Literature review
Chapter 3: ASR and the functionalities of the proposed solution.
Chapter 4: Requirements specification of OLDUM.
Chapter 5: Architecture of the system.
Chapter 6: System Testing.
Chapter 7: System Analysis
Chapter 8: Future work.
Chapter 9: Conclusion
Chapter 10: Bibliography

Chapter 2: Literature Review

2.1 Basics of speech waves

Excitation of vocal cords form changes in air pressure, these waves of changes in air pressure is known as speech or sound. When our vocal cords move it makes the air move back and forth which radiates acoustic energy from the vocal tract. These acoustic signals then cause listener's eardrum to vibrate, thus transforming the acoustic energy into mechanical energy at eardrum. This sound signal is then transmitted as brain signal to the listener's brain which is then processed. Human speech frequency is 85.5Hz – 8.0 kHz and hearing frequency is 50Hz – 20 kHz.

2.2 Speech Recognition System

Speech Recognition is the branch of AI that enables the recognition of speech into text. It can be called as a technology that lets humans talk to a computer. ASR Systems are serving in many ways to automate different speech related processes. These are divided into two different streams i.e., speaker dependent and speaker independent system. Speaker dependent systems only respond to the speakers whose voice is enrolled in it as training data. This is done by making speaker read some text or isolated words. While speaker independent speech recognition system does not rely on vocal training and responds to every speaker.

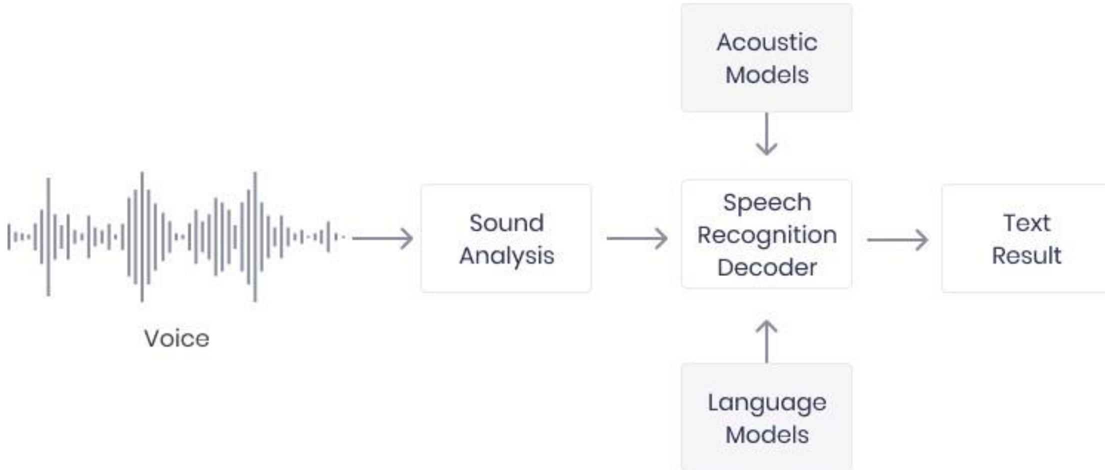


Figure 2.2 General ASR system

2.3 Natural Language Processing

Natural Language Processing is most enhanced version of the Automatic speech recognition systems. It makes humans able to interact with computer through voices and text. Humans use both these forms to communicate. For making our computers understand us in these forms we need to invent ways for making our life easier

2.4 Working of a Speech Recognition

First step is to remove the noise in the audio, then we extract the features from the audio using Mel-Frequency Cepstral Coefficient (MFCC). We pass the audio from several steps for extracting features

OLDUM

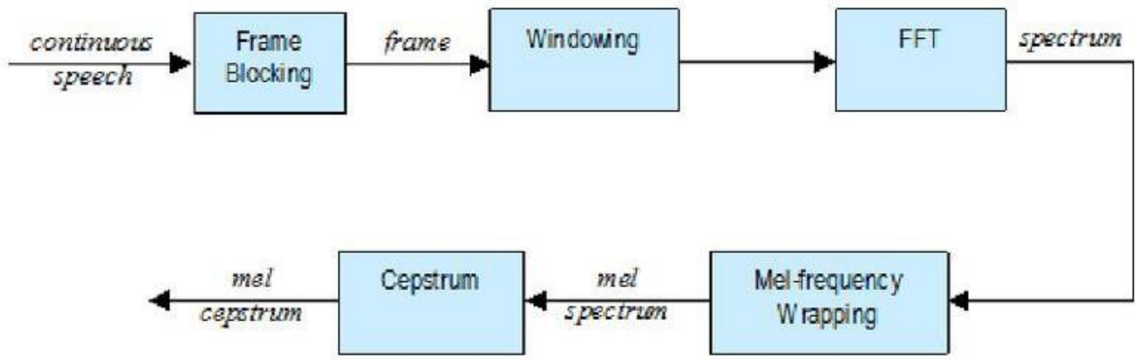


Figure 2.3 MFCC extraction

First we convert the analogue signal to digital signal because computers cannot understand/process analogue data. For this we use Analogue to Digital Converter (ADC). Analogue to digital conversion works in in the following steps sampling, quantization and encoding.

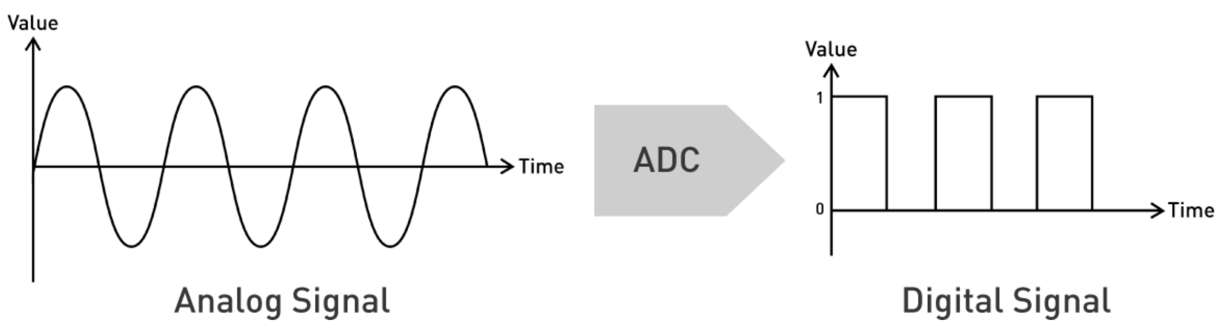


Figure 2.4 Analog to digital

After converting to digital format we apply Fast Fourier Transform (FFT) to get a spectrogram. In this spectrogram, the vertical axis shows frequency, the horizontal axis shows time, and the color

OLDUM

shows the intensity that produces the tone. The brightness/light of the color displays energy used.

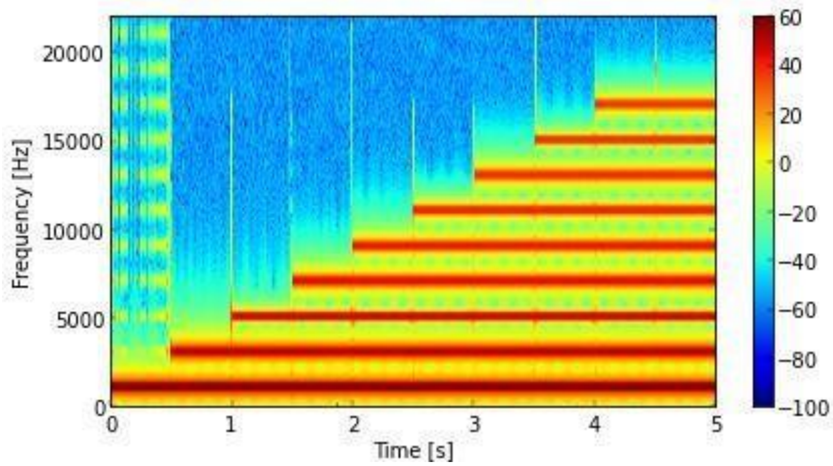


Figure 2.5 Sample of Spectrogram

After getting spectrogram we can extract features, the most important coefficients are the first 12-13 out of total 39 because they contain the most important information such as formants and spectral envelopes.

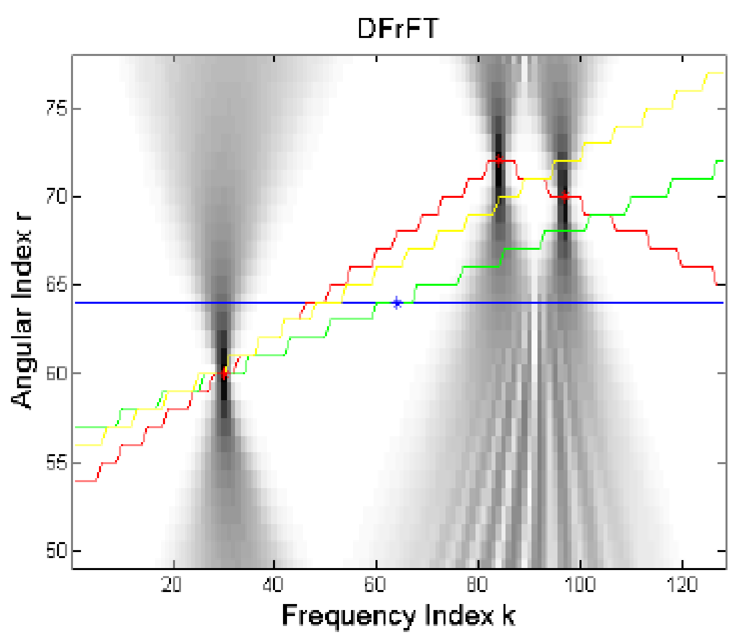


Figure 2.6 Frames

We apply required algorithms required for extracting features

Chapter 3: OLDUM METHODOLOGY

We have included all the techniques in Figure 3.1 because the system is designed to make the entire process as seamless and accurate as possible. After initial processing, the audio signal is sent to the system as an input for further investigation. The feature extraction technique is first performed on the audio signal, as shown below. Prioritized the use of the CNN model for detection on the system. Our model was trained with it. We also used additional models such as the SVM classifier, Nave Bayes algorithm, and random forest classification model to verify accuracy. This paper also briefly describes the characteristics of these models. Training was performed on about 78% of the data to determine the accuracy of the machine learning algorithm. The system results were displayed in the desktop application interface as offensive or non-offensive word labels.

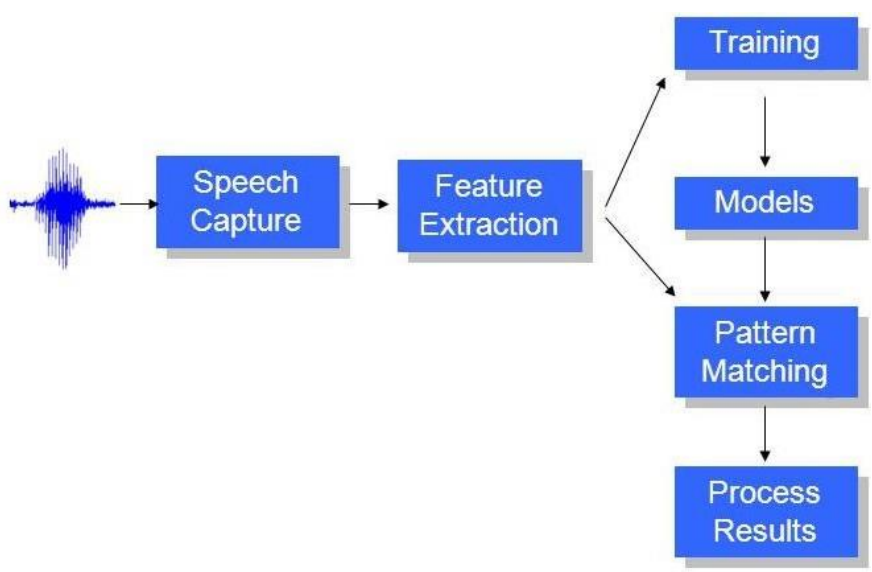


Figure 3.1 Overall Block Diagram

3.1 MFCC Feature Extraction

Before extracting features we need to remove all the unnecessary data known as noise, so we filter the signal to eradicate all the noise. We use feature extraction to illustrate the signal in numbers and evaluate it therefore removing unnecessary details.

We utilize the Mel-frequency cepstral coefficient to depict the short-term power spectrum of a sound in sound processing (MFCC). These are produced from the audio clip's nonlinear spectrum. The MFCC feature extraction approach was also used in our system.

Pre-emphasis, Frame blocking, windowing, taking FFT, then equating the frequencies on a Mel scale, and finally applying inverse DCT are all processes used in the MFCC feature extraction technique. The process block diagram in figure 3.2 gives us an overview of the entire process, which will be detailed in more detail below.

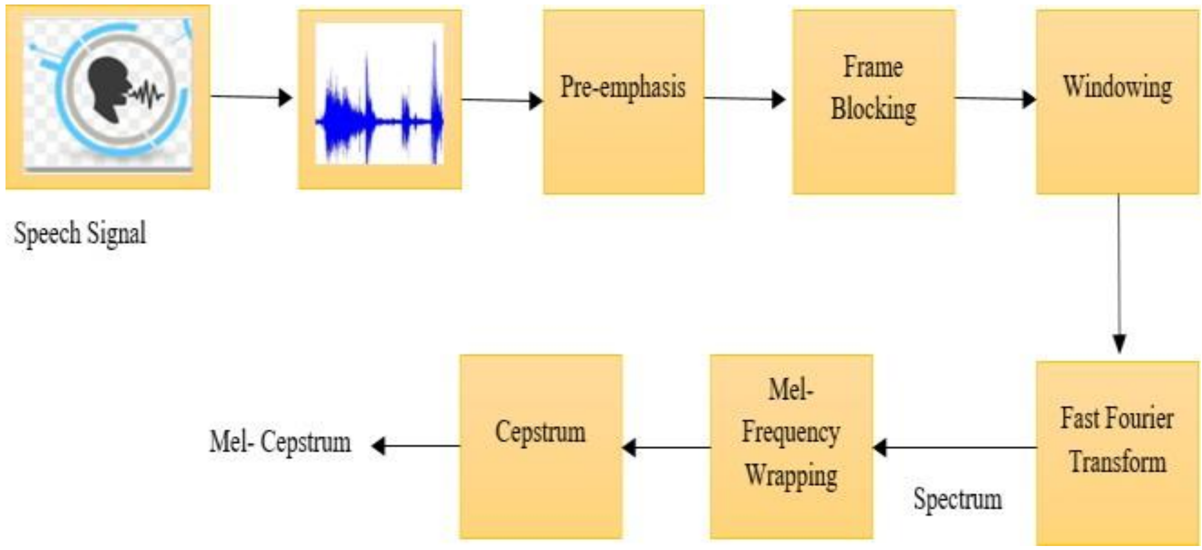


Figure 3.2 MFCC extraction steps

A. Pre-emphasis:

Like applying a high pass filter to make it possible to focus on high frequencies. The glottal effects in the sound are removed with pre-emphasis. The following transfer function is commonly used as a pre-emphasis filter.

$$H(z) = 1 - bz{-1}$$

Where the value of b controls the slope of the filter and is usually between 0.4 and 1.0

B. Windowing and Frame Blocking:

The speech signal is a t Time dependent it must be analyzed for steady acoustic properties over a short period of time. As a result, speech analysis is limited to small chunks of speech that remain fixed. These short-term tests are done in 20 ms or 10 ms periods. This allows for the tracking of individual speech sounds with temporal features, and this window is large enough to provide adequate spectral resolution. Hamming windows are commonly employed in Fourier transforms to increase harmonics and eliminate edge effects.

C. Fourier transform:

The windowed frames are converted to frequency domain by applying Fourier transform

$$Y(n) = \sum_{n=-\infty}^{\infty} x(n)e^{-i\omega n}$$

D. Mel spectrum:

The Mel spectrum is the result of applying a Mel filter bank on the Fourier transformed signal. This is a unit of measurement based on the frequency perceived by the human ear. Below 1 kHz, the Mel scale has a linear frequency spacing, and above 1 kHz, it has a logarithmic frequency spacing.

OLDUM

E. Discrete cosine transform (DCT):

The log Mel spectrum is converted to the time domain using the discrete cosine transform (DCT). The final MFCC characteristics that were required come out of this step. Acoustic vectors are the distinct number of various coefficients created by using DCT. Our system extracted 13 different coefficients [12].

3.2 Machine Learning Algorithms

We have used Convolutional Neural Network (CNN) for our system since because in our research we have found that this is the best one.

3.2.1. Convolutional Neural Network (CNN)

Convolutional neural networks are a machine-learning-based model that uses the convolutional mathematical process. In one of the layers, we employ convolution rather than general matrices multiplication. The input, output, and hidden layers of this neural network are designated accordingly.

Because the activation function and convolution mask the inputs and outputs of this layer, the middle layer is the one that is hidden. Convolution is carried out by these hidden levels. The layer's inputs, in this case the MFCC features and the convolutional kernel, are taken as a dot product. The kernel is then slid across the input matrix, producing a feature map that serves as the input for the following layer. Pooling and normalization layer processes, among others, are also carried out.

In addition to the convolutional layer, there is a pooling layer that minimizes data by mixing the data clusters of one layer's output into a distinct cluster for the next layer. The two most frequent methods of pooling are max pooling and average pooling, which take the maximum value of each cluster and the average value, respectively.

OLDUM

The next step is dropout, which prevents a neural network from overfitting by comparing different probability values for improved accuracy. In the following sections, we will describe the findings for our dataset once the convolutional neural network has been applied to it.

3.2.2 Support Vector Machine (SVM)

A supervised machine learning model is a support vector machine (SVM) [13]. To classify two things, SVM is often used. It is a simple audio that is applied on labelled data. It's more of a binary classifier that can figure out whether a vector belongs in class 1 or 2. The data in our situation had to be classed as offensive or non-offensive. We must assess if our p dimensional plane data can be classified in a p-1 dimensional hyperplane using the hyperplane concept. One of the better hyperplanes is the one that requires us to measure the distance between the two groups. The maximum margin hyperplane has been audited.

The regional data is in finite-dimensional space in certain circumstances, but the sets utilized to discriminate are not in a state where they can be linearly separated in space. This original data has been mapped onto a high-dimensional space that allows us to easily separate it. The dot product of input vectors is simply computed in terms of the original space variables thanks to the usage of a kernel function. A sum of kernels can be used to calculate the distance between each test point and the defined data points.

In terms of the criteria on which it learns the data, SVM is slightly more effective than other methods. The reduction of misclassifications is our primary goal. Depending on the quality of the data, SVM can work on a smaller dataset with higher accuracy. One of the most significant benefits is that it provides a guaranteed one-of-a-kind solution that can even work in noisy conditions [14] [15].

Chapter 4: Software Requirement Specifications

4.1 Introduction

This chapter gives an outline of our system's needs. The purpose of this paper is to provide a full overview of the project to construct an offensive speech tracking device that will automate the tracking procedure. This paper contains the OLDUM requirements in detail.

4.2 Description

4.2.1 Functioning of Product

The following are the key features of OLDUM in this domain:

• Offensive words are recorded by the system.

• Audio files will be provided by users.

• It will listen to audio for objectionable terms in the dataset.

• If the audio contains any foul words, OLDUM will detect them.

• A specialized desktop programme will display the whole procedure.

4.3 User Classes and Characteristics

The sorts of OLDUM users are discussed in the next section.

4.3.1 Security Agencies

Once the prototype is in their hands, security agencies will have access to the system and will be able to enhance the dataset to meet their demands and resources.

4.3.2 Corporate sector

By using relevant statistics, social media platforms will be able to maintain their messages user-friendly while still preserving their privacy.

4.4 Operating Environment

4.4.1 Hardware

OLDUM will have following hardware specifications:

- **Raspberry Pie 4:** For processing sounds as a standalone device.
- **High quality professional condenser microphone:** For recording the dataset/audio manually.
- **LCD:** For display interface of device.

4.4.2 Software

OLDUM will have following Software specifications:

- Linux/Raspbian
- Python IDE

4.5 Design and implementation Constraints

• It will be capable of processing only one audio track at a time.

• The dataset will be limited to a few words and trained on those words, with the ability to increase the dataset as needed.

• The system will require a lot of computing power, therefore we'll have to meet those requirements.

4.6 Documentation for Users

The users will be provided a user manual with instructions on how to run OLDUM as well as the limits that must be considered. Users will also have access to a project report outlining the software's features, capabilities, and procedures.

4.7 Dependencies and Assumptions

• Users must be aware of the dataset's limits and that the system will need to be trained with a different dataset if they want it to be adaptive to their business.

• The system's accuracy will have to be examined.

4.8 External Interface Requirements

4.8.1 User Interfaces

- **Front-end software:** UI based on Raspbian OS / Windows OS for training.
- **Back-end software:** Python

1. Display Screen

The main user interface screen will allow the users to monitor the whole process of offensive speech tracking.

2. Login Prompt Screen

To ensure secure use of the device, a login screen has been put forth that will prompt users to enter their respective credentials to gain access the device.

4.8.2 Hardware Interfaces

- Windows Operating System (For training the machine)
- Raspbian OS for Raspberry Pie (For Implementation).

4.8.3 Software Interfaces

Software used	Description
Operating system	We chose Windows 10 as our operating system because of its excellent support and user-friendliness. Raspbian OS will be used for the end user interface.
Database	The user information will be maintained in a database, and users will be prompted to input their credentials every time they visit.
Python	Training the machine and other procedures are generally simpler and more efficient so we chose Python as our main programming language.

Table 4.1 Software Interfaces

4.9 Communication Interfaces

Our system uses Python in the background to process (for inappropriate terms) every audio file uploaded or listened to in real time, as well as a UI based on the Raspbian OS to provide real-time feedback. In order to secure the entire process and system, a login/signup option has been introduced.

4.10 System Features

OLDUM will be providing following system features:

1. Signup

- **Description and Priority**

OLDUM

User will first signup/register to OLDUM to get access to all system features. It is of medium priority.

- **Stimulus/Response Sequences**

1. The user will open the software system.

2. The user will then switch to signup screen.

3. User will enter registration details to sign up for using OLDUM.

- **Functional Requirements**

Req 1: Users should be able to access signup page.

Req 2: User should be able to enter registration details in proper fields.

Req 3: System should be able to register new users properly.

2. Login

- **Description and Priority**

User will first login to OLDUM for further processing on audio files and detecting offensive audios. It is of medium priority.

- **Stimulus/Response Sequences**

1. The user will open the software system.
2. The user will then switch to login screen.
3. User will enter login details to access OLDUM.

- **Functional Requirements**

Req 1: Users should be able to access login page.

Req 2: User should be able to enter login details in proper fields.

Req 3: Only authorized users should be able to login.

3. **Receive Audio File**

- **Description and Priority**

OLDUM allows the system to receive & upload different audio files for further processing. It is of high priority because without the audio file, the system will be useless.

- **Stimulus/Response Sequences**

1. The user will upload already recorded audio files from memory disk.
2. For real-time processing, microphone or direct audio audios can be used as input.

- **Functional Requirements**

Req 1: User should be able to upload the audio file in a specific format (Wav.).

Req 2: User should be able to upload audio of up to a specific file size.

4. **Classify audio content/words as offensive or not offensive**

- **Description and Priority**

The main feature or the main purpose of OLDUM is to classify the audio words as offensive or not. It is of high priority because that is the only purpose of developing OLDUM.

- **Stimulus/Response Sequences**

1. The user will upload the audio files.
2. The system will match audio words with already stored offensive words in the dataset.

- **Functional Requirements**

Req 1: The system should be able to match the uploaded audio with the already given dataset.

Req 2: The system should be able to correctly distinguish between offensive and non-offensive words.

5. **Alert the user for offensive audio**

- **Description and Priority**

OLDUM

OLDUM will immediately alert the user if it detects any offensive words in the audio file or audio. It is of very high priority because the user needs to be alerted of the offensive audio to act immediately to stop the ill activities.

- **Stimulus/Response Sequences**

1. The user will upload audio files.
2. The system will match the audio words with the offensive words in the dataset.
3. Upon detecting offensive words, the user will be alerted on the screen for offensive audio.

- **Functional Requirements**

Req 1: The user should be able to receive proper alerts for offensive words.

Req 2: The UI should display a proper message to the user on detecting the offensive audio.

6. Update Dataset/Offensive words list

- **Description and Priority**

User will be able to update OLDUM for new offensive words. The priority is low since it depends on the need of user and is not the main requirement.

- **Stimulus/Response Sequences**

1. The user will add more offensive words to the dataset.
2. The user will then train the system to detect new offensive words.
3. The user will test the system to detect the new offensive words from the audio files.

- **Functional Requirements**

Req 1: Users should be able to successfully add new offensive words to dataset.

Req 2: User should be able to test the system for detecting these new offensive words.

7. Sign out

- **Description and Priority**

User will logout of OLDUM when not in use to prevent its usage by unauthorized person. It is of low priority because logging out depends on the need of user.

- **Stimulus/Response Sequences**

1. The user will click on the logout button.

- **Functional Requirements**

Req 1: Users should be able to access logout button.

Req 2: User should be able to logout successfully without any loss.

4.11 Non-Functional Requirements

4.11.1 Performance Requirements

- High processing power operating systems must be able to run the application.
- The accuracy of the system should be good enough to make it a trusted system.

4.11.2 Security Requirements

- Only authorized users can access the system to prevent any unwanted changes in it.

4.11.3 Software Quality Attributes

- **Availability:** Whenever the user wants to access the software, he can access it if the hardware and software requirements are fulfilled.
- **Maintainability:** The dataset should be maintained by adding new offensive words.
- **Reliability:** The system shall have a higher level of accuracy which would ensure its reliability, our target was to get at least 85 % accuracy which we achieved.

Chapter 5 : Design and Development

5.1 Introduction

This chapter focuses on the work breakdown structure of our system into small deliverables, as well as the system architecture and design based on all of the functional requirements outlined in the previous chapter.

5.2 Work Breakdown Structure

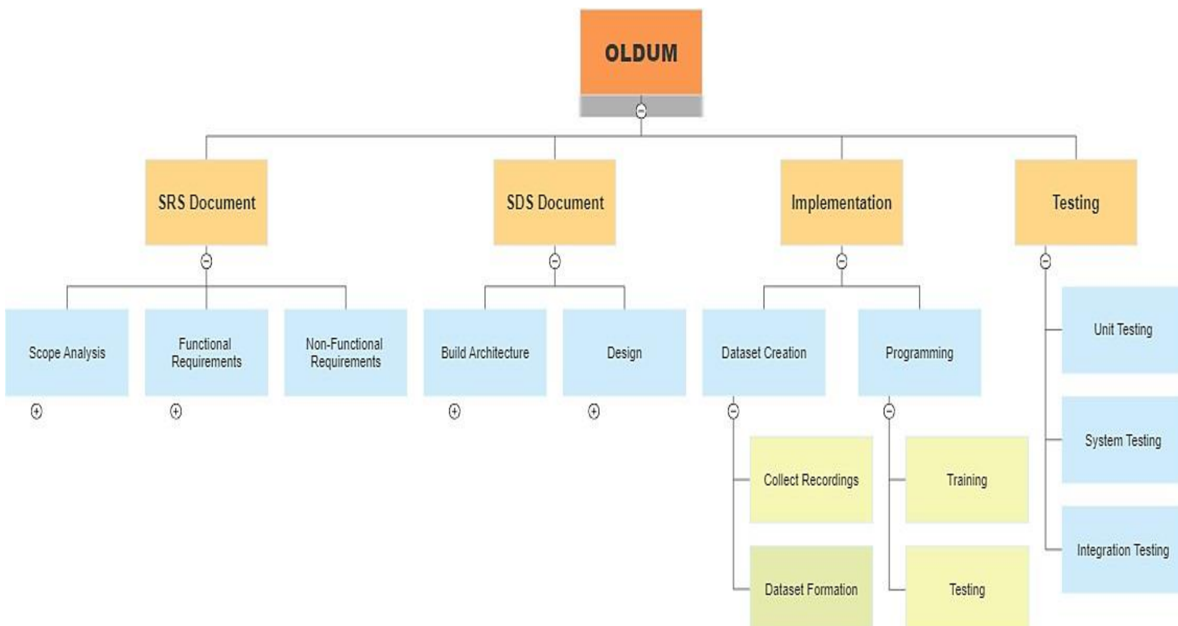


Figure 5.1 WBS

5.3 Architectural Design

Provided are the modules of our prototype that will be followed:

Figure 5.2 Block Diagram

5.4 Decomposition Description

The diagram(s) show the higher-level description of the application(s), generic working of the application(s) and interaction with the user.

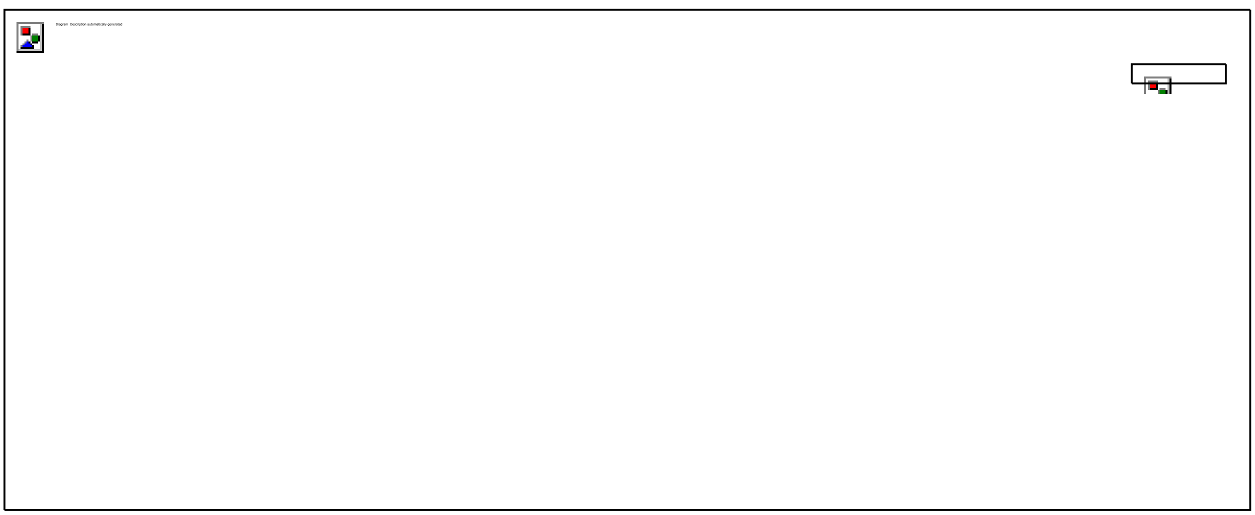


Figure 5.3 Class diagram

5.5 Use Cases

A use case is an approach for locating, clarifying, and organizing system needs in the context of system analysis. A use case is a collection of possible interactions between systems and users in a given environment that are all related to a single goal.

The various user classes recognized the following use cases and significant actors for OLDUM:

Actors	**Use Cases**
Admin	• Login • Manage Users • Add Recordings • Start Tracking • Log out
Users	• Login • Add Recordings • Start Tracking • Log out

The aforementioned use cases can be shown as the use case diagram given below.

5.3.1 Use Case Diagram

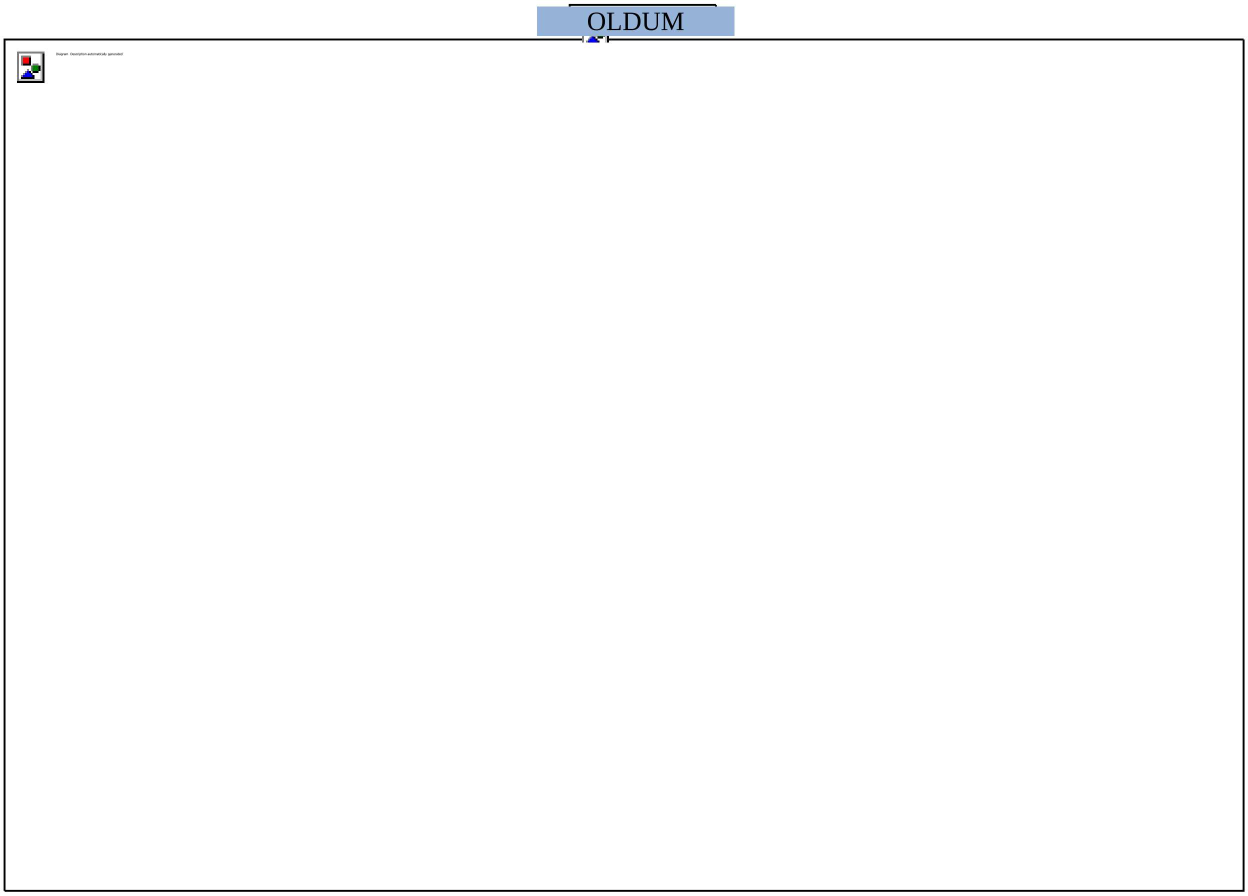


Figure 5.4 - Use Case Diagram

5.3.2 Use Cases Description

1. Login

User ID:	1		
Username:	Login		
Access:	Admin, Users		
Created by:	Ijaz Shah	Last Updated by:	Ijaz Shah
Date Created:	14/03/2022	Date Last Updated:	14/03/2022
Description:	A user tries to login to the system.		
Preconditions:	User has to open the login page first.		

Post conditions:	If the use case was successful, the actor is logged into the system. If a user enters an incorrect username and password, the system state remains unchanged, and the user is prompted to re-enter his credentials.
Normal Flow (Primary Scenario):	1. The performer is prompted to input his or her private credentials, such as a user name and password. 2. The performer logs in with his or her account and password. 3. The system checks the actor's account and password and logs them into the system.
Alternative Flows:	If the actor leaves the fields blank or enters an invalid login and/or password in the Basic Flow, the system displays an error message. The actor can either return to the beginning of the Basic Flow or cancel the login, at which point the use case will end.

Table 5.1 Login use case description

2. Add Users

Use Case ID:	2		
Use Case Name:	Add Users		
Actors:	Admin		
Created by:	Hamza	Last Updated by:	Hamza
Date Created:	14/03/2022	Date Last Updated:	14/03/2022
Description:	Admin must log in to the system and add new users to allow them access to OLDUM.		
Preconditions:	Admin must log in.		
Post conditions:	New users should be updated in the database.		

Normal Flow (Primary Scenario):	1. The administrator will log in and go to the administrative panel. 2. The administrator will create new users. 3. The database will be updated with the changes.
Alternative Flows:	1. During the database modification process, an error occurs. 2. The database's functionality will be examined.

Table 5.2. Add user use case description

3. Remove Users.

Use Case ID:	3		
Use Case Name:	Remove Users		
Actors:	Admin		
Created by:	Hamza	Last Updated by:	Hamza
Date Created:	14/03/2022	Date Last Updated:	14/03/2022
Description:	Admin must log in to the system and remove users to disallow access to OLDUM.		
Preconditions:	Admin must log in.		
Post conditions:	User should be removed from the database.		
Normal Flow (Primary Scenario):	1. The admin will login and switch to Admin panel. 2. The Admin will delete users. 3. Changes will be updated in the database.		
Alternative Flows:	1. An error is encountered during the modification of database. 2. Proper functionality of the database will be checked.		

Table 5.3 remove user use case description

4. View Users

Use Case ID:	4		
Use Case Name:	View Users		
Actors:	Admin		
Created by:	Ijaz	Last Updated by:	Ijaz
Date Created:	14/03/2022	Date Last Updated:	14/03/2022
Description:	Admin must log in to the system and view the registered users.		
Preconditions:	Admin must log in.		
Post conditions:	List of registered users should be displayed.		
Normal Flow (Primary Scenario):	1. The admin will login and switch to Admin panel. 2. The Admin will click view users. 3. List of users will be displayed.		
Alternative Flows:	1. An error is encountered while accessing the database. 2. Proper functionality of the database will be checked.		

Table 5.4 remove user use case description

5. Add Recordings

Use Case ID:	5		
Use Case Name:	Add recordings		
Actors:	Users		
Created by:	Hasnat	Last Updated by:	Hasnat
Date Created:	14/03/2022	Date Last Updated:	14/03/2022

Description:	When Users login to their profile, they will be provided access to OLDUM. They can add new recordings to track out offensive words from them.
Preconditions:	User must log in.
Post conditions:	Audio name will be displayed on the screen on successful upload.
Normal Flow (Primary Scenario):	1. The user will login to the profile. 2. User can add recordings.
Alternative Flows:	1. An error is encountered while adding recordings. 2. System will ask user to re-enter recordings.

Table 5.5 Add recordings use case description

6. Start Tracking

Use Case ID:	6		
Use Case Name:	Start Tracking		
Actors:	OLDUM		
Created by:	Hasnat	Last Updated by:	Hasnat
Date Created:	14/03/2022	Date Last Updated:	14/03/2022
Description:	After Uploading audio, user must click Start Tracking button for OLDUM to analyze the audio.		
Preconditions:	User must log in and upload an audio.		
Post conditions:	OLDUM will analyze the audio.		
Normal Flow (Primary Scenario):	1. The user will login and add recordings to be tracked in the system. 2. The user will click Start Tracking button.		
Alternative Flows:	-		

Table 5.6. Start tracking use case description

7. Log Out

Use Case ID:	7		
Use Case Name:	Log Out		
Actors:	Admin, Users		
Created by:	Zeeshan	Last Updated by:	Zeeshan
Date Created:	14/03/2022	Date Last Updated:	14/03/2022
Description:	Actor attempts to log out of the system.		
Preconditions:	Actor must be logged in first.		
Post conditions:	The Actor will be logged out and sent to the login page.		
Normal Flow (Primary Scenario):	1. Actor clicks the Sign Out button. 2. The Actor is signed out and sent to the Login Screen.		
Alternative Flows:	-		

Table 5.7 log out use case description

5.6 Sequence Diagrams

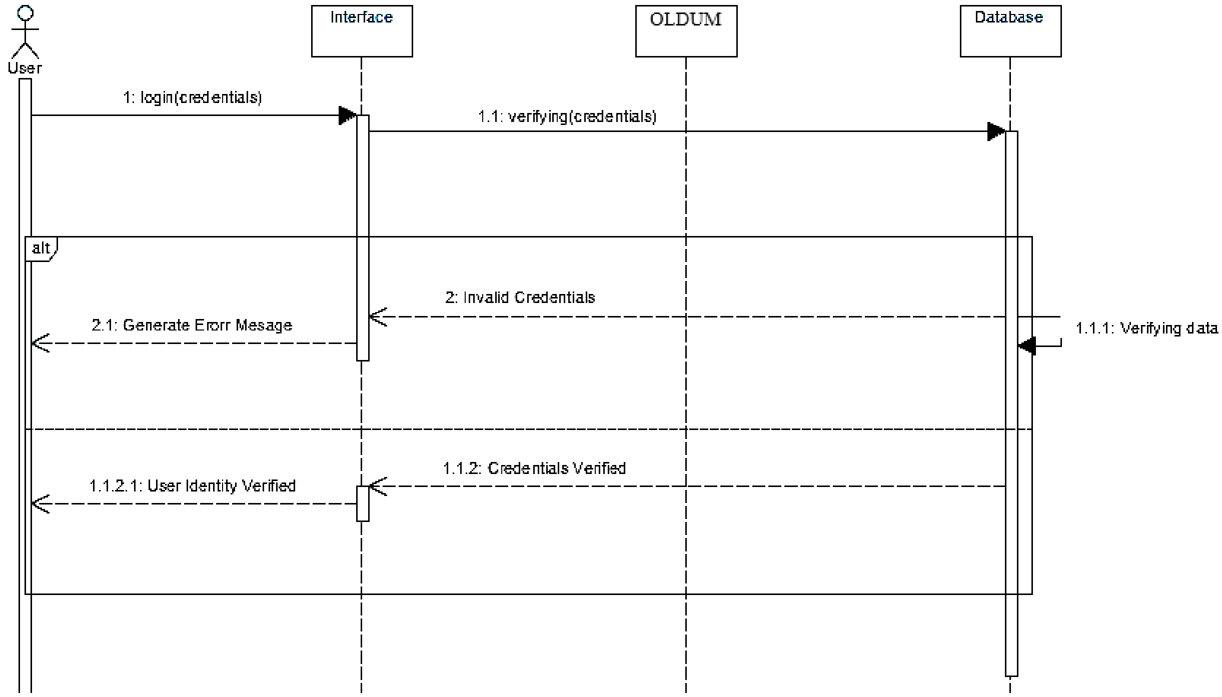


Figure 5.5 – Login

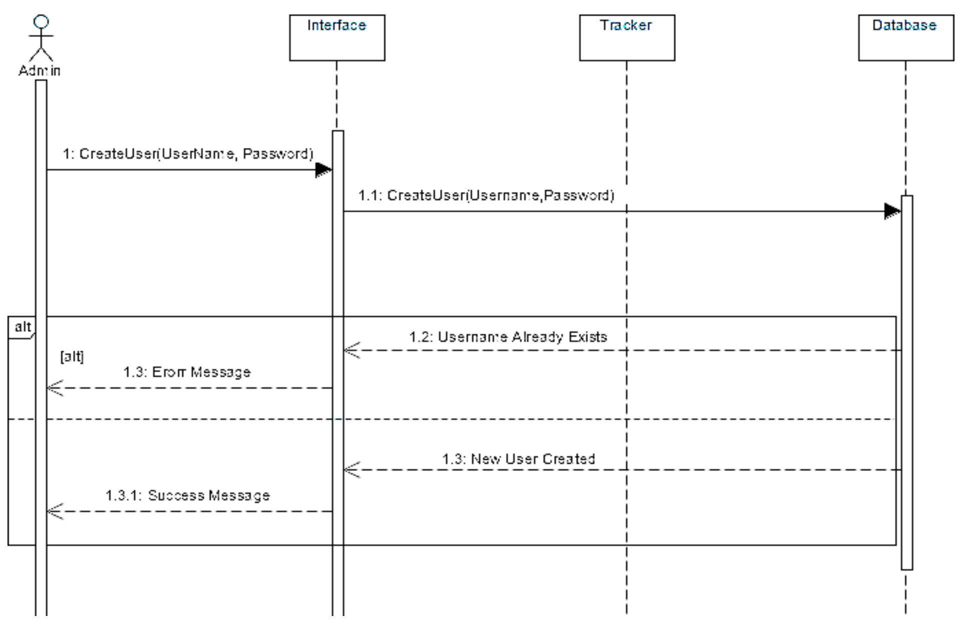
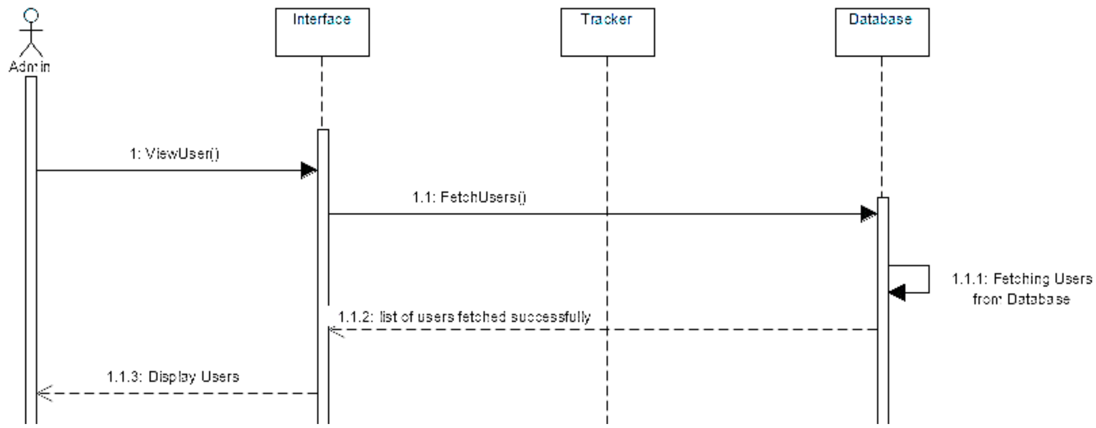
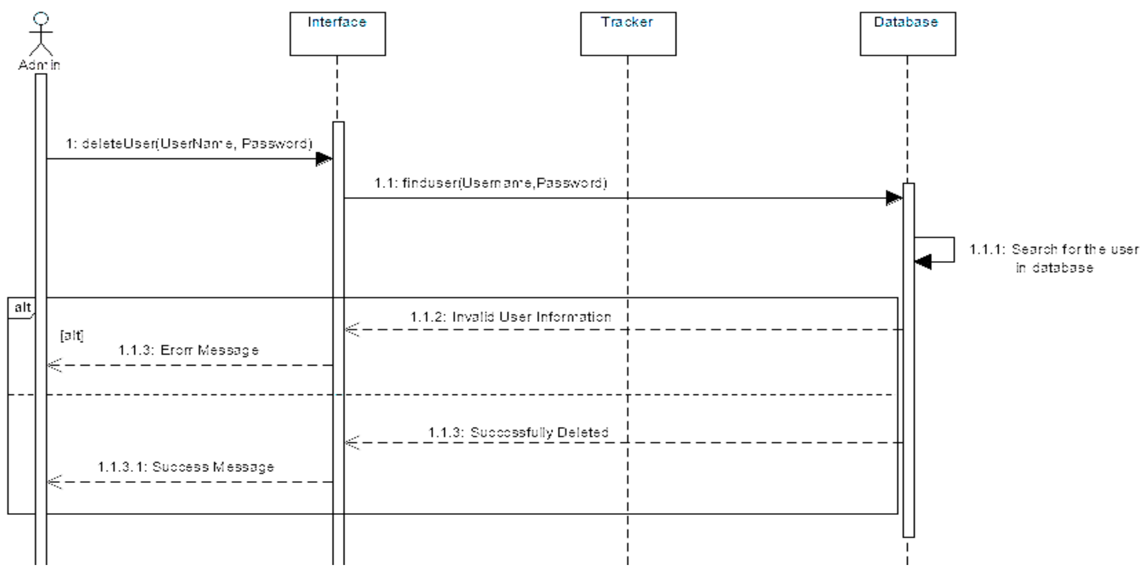


Figure 5.6 Add User





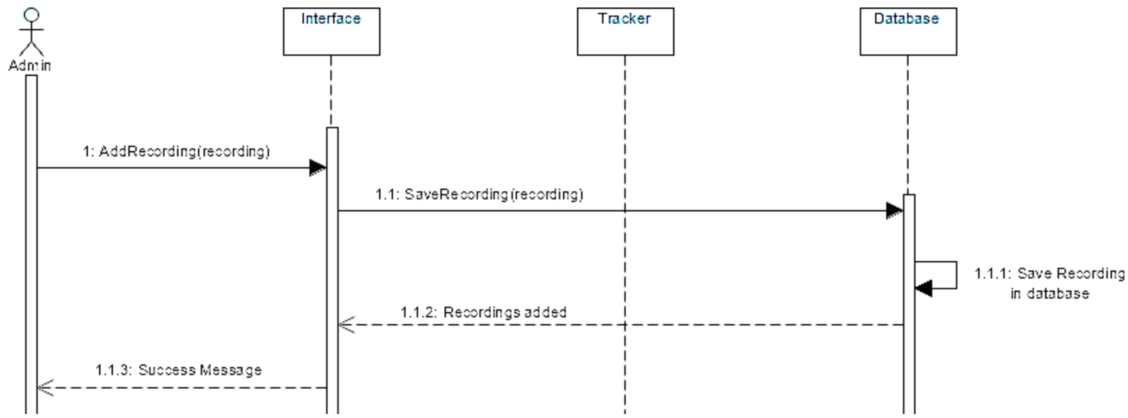


Figure 5.9 Add Recordings

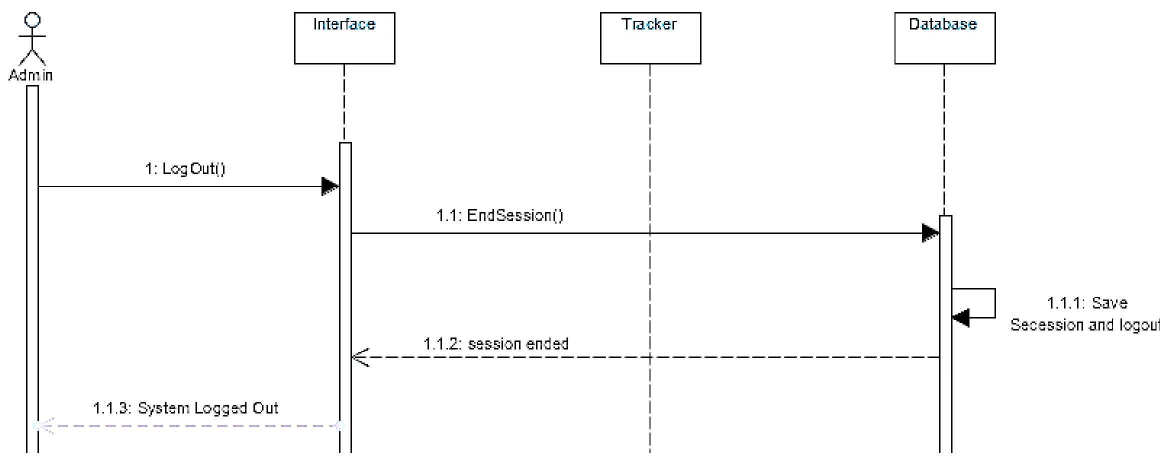


Figure 5.10 Logout

5.7 Data Design

5.7.1 Data Description

System saves user data in the OLDUM database including username and password. The database that is used is **MySQL**. Audio recordings will also be processed by the system and be tracked for offensive words. The audio uploaded by user will be matched with already provided dataset in the files. If any offensive words will be found, then the user will receive an alert message.

5.7.2 Data Dictionary

Objects	Attributes	Methods	Parameters
Admin	loginID: String Password: String AudioRecording:audio	AddUser()	Input: String, String
		RemoveUser ()	Input: String
		ViewUser()	Input: String
		SignIn()	Input: String, String
		SignOut()	
		AddRecording()	Input: Audio
Database OLDUM	Admin: String User: String	adminAuthentication()	Input: String Output: boolean
		userAuthentication()	Input: String Output: boolean
General User	loginID: String Password: String AudioRecording:audio	SignIn()	Input: String,String
		SignOut()	
		AddRecording()	Input: Audio
		StartTracking()	Input: Audio
Tracker		TrackRecording()	Input: Audio
		MarkRecordingAsOffensive()	Output: boolean
		SendAlertMsg()	Output:String

Table 5.8 Data dictionary

5.8 Component Design

1. Login

- Application displays a login button.
- User clicks the login button
- A prompt is displayed to input login credentials.
- User enters username and password and clicks login button.
- If credentials are correct, user will enter the system, else error message is displayed.

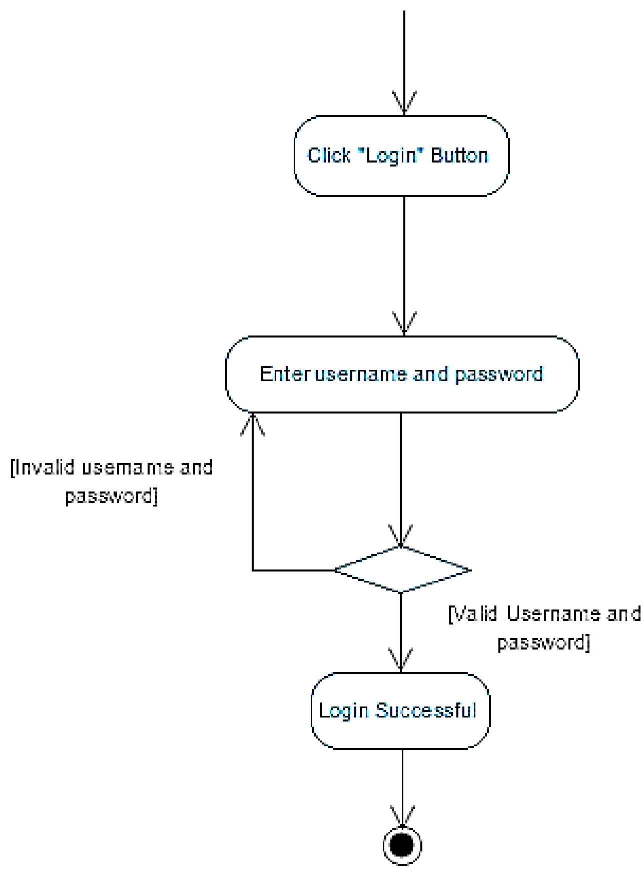


Figure 5.12 Login

Add user

- Admin logs in and click on add user button.
- A prompt is displayed to input credentials for new user.
- Admin enters username and password for new user.
- If username is already in use error message will be displayed, else new user profile is created.

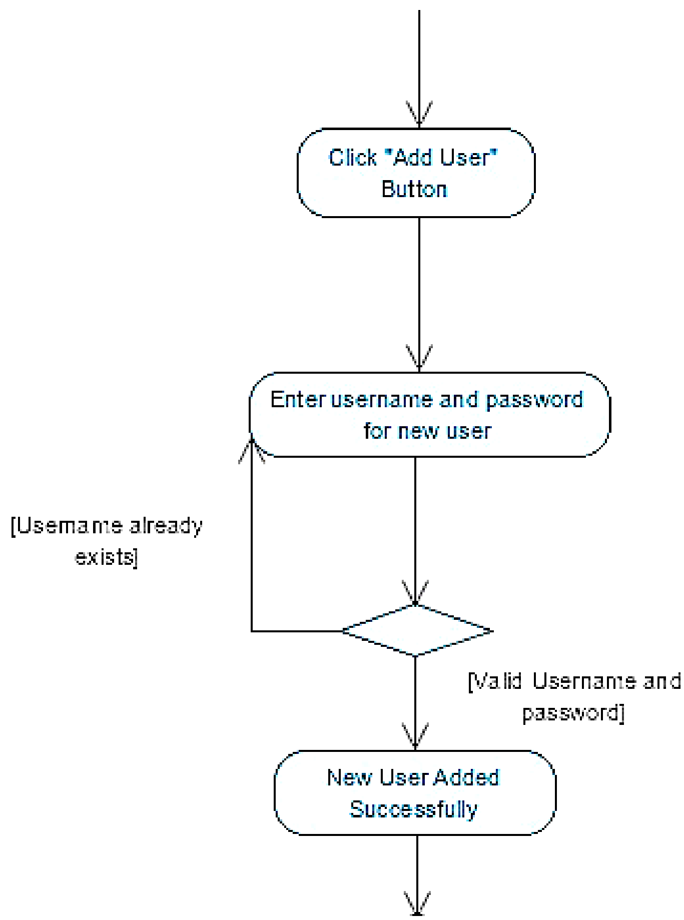


Figure 5.13 Add User

View User

- Admin logs in and click on view users’ button.
- List of all the users with all their relevant information is displayed on the screen.

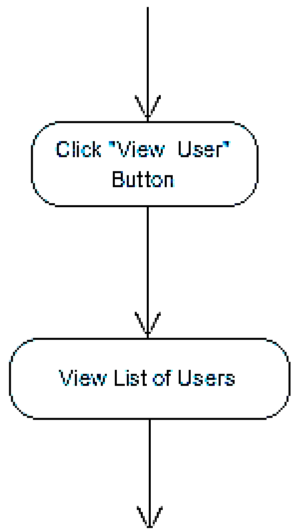


Figure 5.14 View Users

Delete User

- Admin logs in and click on remove user button.
- A prompt is displayed to input credentials of user to be removed.
- Admin enters username and password for user.
- If entered username doesn’t exist error message will be displayed, else user will be deleted from the system.

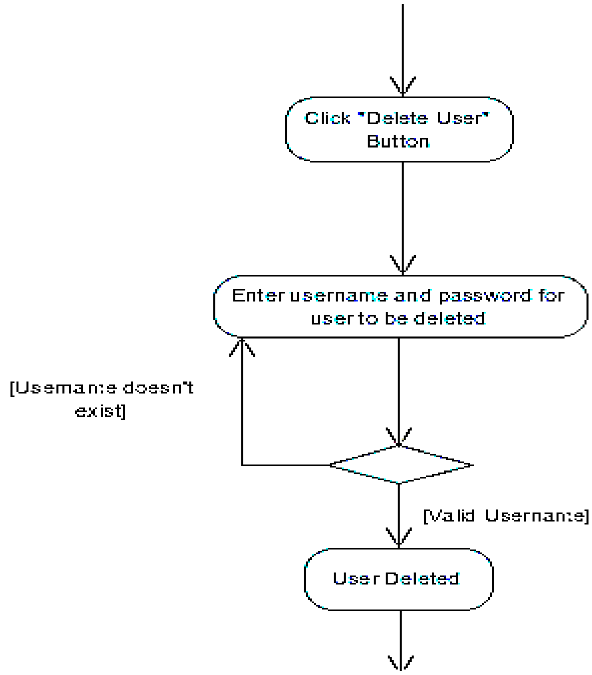


Figure 5.15 Delete User

Add Recordings

- User logs in and click add recordings.
- User adds recordings to be tracked.
- If added recordings are of specific format and size, it will be uploaded, else error is generated.

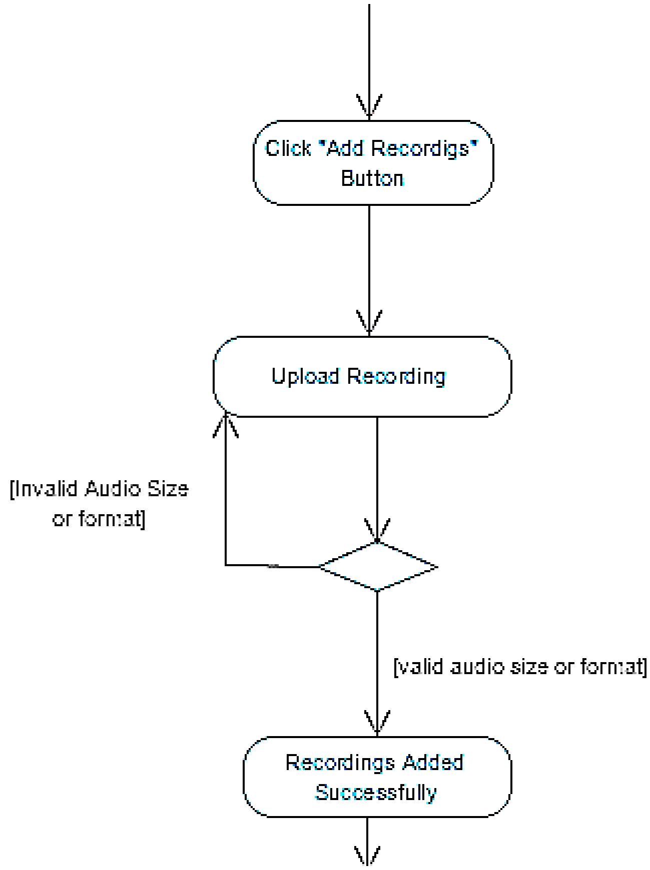


Figure 5.16 Add Recordings

Track Recordings

- Admin logs in and click on track recording button.
- List of uploaded recordings will appear.
- User will select the recording to be tracked.
- If the recording contains any offensive word alert message will appear on the screen, else a prompt will appear declaring the recording to be safe.

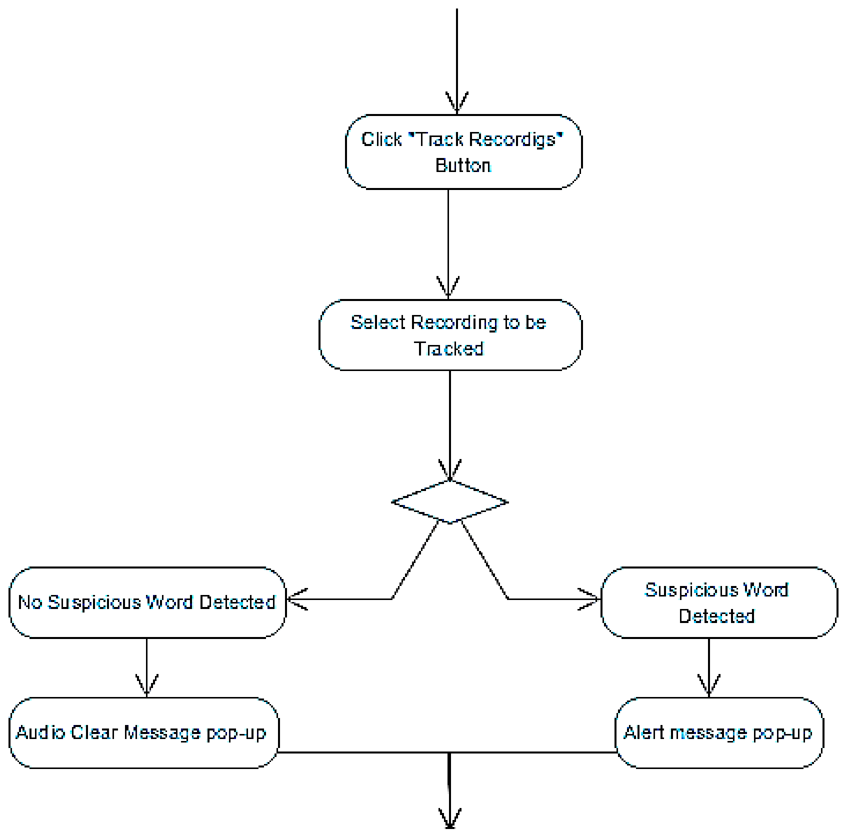


Figure 5.17 Track Recording

Log Out

- User will login to the system.
- After tracking its required recordings, user will click on log out button.
- Session information will be saved and user will be logged out of the system

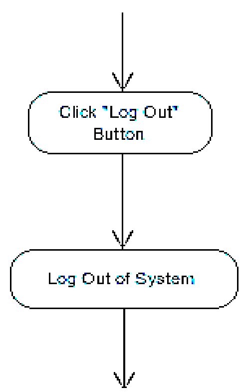


Figure 5.18 Log Out

5.9 Human Interface Design

5.9.1 Overview of User Interface

The users of the system will be able to login to the OLDUM system and then upload an audio to start the tracking process. Only the admin of the system will be allowed to add new users, remove users, and view users. Additionally, admin can upload an audio and start its tracking. The main page of OLDUM desktop application consists of two buttons: upload audio, start tracking. There is one login page for users to be able to access the system functionalities. The admin account will have an additional page/ admin panel to be able to add, delete and view users.

5.9.2 Screen Objects and Actions

1. Login:

Users and Admin can login with his username and password. The login form is validated with incorrect username or password, empty username or password and then successful login for correct username and password. Following buttons or fields are on this page:

1. Username field
2. Password field
3. Keep me logged in button
4. Login button

2. Home page:

After logging in, users are directed to the home page from where user can access the option of uploading an audio and start its tracking. Once the Start Tracking button is pressed, OLDUM will start analysing the uploaded audios and match the words with dataset. If a offensive word is found in the audio, user will be immediately alerted. Following buttons are on this page:

1. Upload Recording button
2. Start Tracking button

3. Admin panel:

The admin panel gives the admin access to adding new users, deleting users, and viewing users from the database. No other user will have access to these functionalities. Following buttons are on this page:

1. Add User button
2. Delete User button
3. View Users button

Chapter 6 : Testing

6. Analysis and Evaluation

6.1. Introduction

This portion will cover the methodologies used to execute and manage "OLDUM" testing process. The test plan will ensure that OLDUM is able to fulfill all the requirements.

6.2. Approach

All the test cases will be stated first and then analyzed based on a report that will then be summarized to check for the OLDUM performance.

We will be using these steps:

1. **Unit test**: Here every single module has to be verified individually in order to check for performance.
2. **Integration test:** The second step after the unit test is to implement the integration test cases where all the individual portions are to be integrated with each other and tested.
3. **Positive and negative testing design technique:** Here we must design specific test cases in certain situations. This will in turn show if all the functional requirements are being fulfilled. Invalid situations will also be tested to see if the system can judge them.

6.3. Features to be tested

Following Features are tested:

1. Only authorized admin/users should be able to log in to the system.
2. Admin should be able to add a new user.
3. Admin should be able to view all users.

4. Admin should be able to remove users.
5. Users/Admin should have eased to add recordings in a given format and size.
6. Users/ Admin should be able to track recordings.
7. OLDUM should be able to detect offensive audios.
8. The user should be able to receive proper alerts for offensive words.
9. Admin/Users should be able to log out of the system.

6.4. Pass/Fail Criteria

Individual details of test cases are mentioned in section 6.11. Here are some criteria to judge these tests.

1. Prerequisites are met
2. Inputs are accepted
3. Output is as it was expected to work

6.5. Testing tasks

1. Test cases to be developed
2. Tests to be executed as given
3. Results to be compiled from these cases.
4. Cover the changes required in the system during upgradation.

6.6. Test Deliverables

- Test cases
- Output from tools

6.7. Responsibilities:

Group members are responsible to carry out the testing be it the individual components or in the integration part.

6.8. Staffing and Training Needs:

Basic expertise is needed in testing the system for example Black Box testing, integration testing etc. Active participation as a team will be required.

6.9. Schedule

6.9.1. Important Dates

1. Unit Testing and integration testing will be finished by the start of 28 May 2021 as will Development process
2. Acceptance Testing will be performed right after the Development process completes that is in the start of June.

6.10. Risks and contingencies

6.10.1. Schedule Risk:

Proper WBS should be followed as given in the document along with proper management of time or the project will be at risk of falling behind schedule.

6.10.2. Operational Risks:

Proper meetings and goals should be designed in accordance with schedule in order to make sure that testing goes according to plan. There should not be any communication gap.

6.10.3. Technical risks:

Expertise of group members will be taken in to account in order to minimize technical risks.

6.10.4. Programmatic Risks:

The scope of the project will be kept in accordance with the degree limitations to manage this.

6.11. Test Cases

6.11.1. Unit and Component level Testing

Test Case Number	01
Test Case Name	Admin/User Login
Description	Only authorized admin/users should be able to login to the system.
Testing Technique	Component testing, Black Box Testing
Preconditions	Web portal of OLDUM should be open
Input Values	Enter username and password then click “Login”
Steps	1. The user will open the software system. 2. The user will then switch to login screen. 3. User will enter valid login credentials to access OLDUM.
Expected output	Admin/user should be logged in.
Actual output	Admin/user is logged in successfully.
Status	Test case successfully passed.

Table 6.1 Admin/User Login

Test Case Number	02
Test Case Name	Add new user
Description	Admin should be able to add new user.
Testing Technique	Component testing, Black Box Testing
Preconditions	Web portal should be open and admin should be already logged in to the system.

Input Values	Add new users credentials
Steps	1. Open the web portal. 2. Login as admin. 3. Click add new user. 4. Enter new user credentials 5. Access the 'Add user' option.
Expected output	New user credentials should be registered to the database.
Actual output	Successfully registration of user.
Status	Test case passed successfully.

Table 6.2 Add New User

Test Case Number	03
Test Case Name	Viewing all the users
Description	Admin should have the access to see all the users operating OLDUM.
Testing Technique	Unit testing, Black Box Testing
Preconditions	Web portal should be open and admin should be already logged in to the system.
Input Values	Press "View All Users"
Steps	1. Open the web portal. 2. ADD admin ID. 3. Access 'Login' option. 4. Open "View All Users"
Expected output	Admin should have the access to see all the users operating OLDUM.
Actual output	Admin can successfully view all users.
Status	Test case passed successfully.

Table 6.3 View All Users

Test Case Number	04
Test Case Name	Remove User
Description	Admin should be able to remove users.
Testing Technique	Component testing, Black Box Testing

Preconditions	Web portal should be open and Admin should be successfully logged in.
Input Values	Press the “Delete user” button and add valid credentials for the user to be deleted.
Steps	1. Open the web portal. 2. Login as Admin. 3. Press the “Delete user”. 4. Enter credentials for the user to be deleted.
Expected output	User should be deleted from database.
Actual output	User is successfully deleted.
Status	Test case passed successfully.

Table 6.4 Remove User

Test Case Number	05
Test Case Name	Add Recordings
Description	Users/Admin should have ease to add recordings in a given format and size.
Testing Technique	Component testing, Black Box Testing
Preconditions	Web portal should be open and Admin/User should be successfully logged in.
Input Values	Press “Add Recording” and upload recording from your system.
Steps	1. Open the web portal. 2. Login as admin/user. 3. Click “Add Recording”. 4. Upload recoding in
Expected output	Admin/Users should be able to add recordings if the size and format is correct.
Actual output	Recording successfully added.
Status	Test case passed successfully.

Table 6 5 Add Recordings

Test Case Number	06
Test Case Name	Track Recordings
Description	Admin/Users should be able to track recordings.

Testing Technique	Component testing, Black Box Testing
Preconditions	User/Admin should be logged in and has uploaded recording.
Input Values	Click on "Track Recording".
Steps	1. Open web portal. 2. Login as Admin/User. 3. Click on "Add Recording". 4. Click on "Track Recording".
Expected output	Recordings should be tracked.
Actual output	Recordings tracked successfully.
Status	Test case passed successfully.

Table 6.6 Track Recordings

Test Case Number	07
Test Case Name	Detect Offensive Audios
Description	The system should be able to detect offensive audios.
Testing Technique	Component testing, Black Box Testing
Preconditions	Admin/User is logged in. Admin/User adds new recordings and clicks on track audio.
Input Values	Click on track audios and wait for the system to track it.
Steps	1. Open web portal. 2. Login as Admin/User. 3. Click on "Add Recording". 4. Click on "Track Recording". 5. System will track the audio.
Expected output	System should track the audio.
Actual output	Audio is tracked successfully by the system.
Status	Test case passed successfully.

Table 6.7 Detect Offensive Audios

Test Case Number	08
Test Case Name	Alert Message

Description	The user should be able to receive proper alerts for offensive words.
Testing Technique	Component testing, Black Box Testing
Preconditions	Admin/User is logged in. Admin/User tracks the audio.
Input Values	System tracks the audio file.
Steps	1. Open web portal. 2. Login as Admin/User. 3. Click on "Add Recording". 4. Click on "Track Recording". 5. System will track the audio. 6. System will show an alert message if the audio is offensive.
Expected output	Alert message should pop up.
Actual output	Alert message pops up.
Status	Test case passed successfully.

Table 6.8 Alert Message

Test Case Number	09
Test Case Name	Logout
Description	Admin/Users should be able to logout of the system.
Testing Technique	Component testing, Black Box Testing
Preconditions	User/Admin is logged in.
Input Values	Click on "Log Out" button.
Steps	1. Click on 'Account Settings' 2. Enter edited information 3. Click on 'Submit Form'
Expected output	Admin/Users should logout of the system.
Actual output	Admin/Users logouts of the system.
Status	Test case passed successfully.

Table 6.9 Logout

Chapter 7 : Analysis

OLDUM was created with the goal of making automatic speech recognition for offensive speech tracking more user-friendly. At the same time, accuracy must be considered in such systems because it is of paramount importance. To achieve the best possible accuracy, a rigorous quality and accuracy check was performed on our dataset on a regular basis.

We used our Pashto-based recognition system here, which included 15 words: 10 non-offensive and 5 offensives. The training and testing sections of the dataset were split into two pieces. After extracting features with the MFCC approach, machine-learning algorithms were applied to the data.

We experimented with four different algorithms to see which one best supported our system. For the readers' convenience, we've highlighted the accuracies for all of the systems to make it easy to compare how different models perform in similar situations. The table below shows the accuracy of all of the algorithms.

Algorithm	Accuracy
CNN	86%
Random Forest Classification	70.2%
RNN	65%
SVM	74%

Table 7.1 Accuracy of Algorithms

The best accuracy was achieved by the **Convolutional Neural Network**, as evidenced by the preceding results. CNN took the shortest time. As a result, we used CNN in our system.

We discovered that using various methodologies for our system, CNN may be quite effective in learning features directly from the dataset during the training process. We utilized MFCC to tune CNN in this case, and it worked perfectly. Our system's results show that a well-trained Convolutional Neural Network maximizes the system's performance. Previous methodologies were found to be improved and added value by our proposed system.

In addition, we examined all 15 words in tables 2 and 3 for four different speakers to determine our system's word mistake rate. We explored several speakers while using the same training and testing data to attain the lowest WER. The percentage inaccuracy per word of each speaker is shown in the table below.

Test Number	% Error
1	0
2	0
3	13.34
4	13.34

Table 7.2 CNN error rate

From the above results we conclude that the total percentage error of our system is 13.34%. This could be further reduced by adding more data to our dataset, which is a goal to expand our prototype system.

A specific desktop application with a very user-friendly interface was also designed and connected with our technology for easy audio tracking. This application is up and running, and users can upload audio files for analysis utilizing the OLDUM system on the backend. This application's accuracy was similarly comparable to what we provided in our thesis, making it a useful platform for tracking offensive speech.

Chapter 8 : Future Work

OLDUM can be enhanced to automate the voice tracking procedure for big word sets. To include as much Pashto language as feasible, we will endeavor to contribute more data marked as offensive and non-offensive to our project. The technology could attain greater accuracy with more data. Furthermore, to rely on it as an accurate method, we must cover all of Pashto's conceivable vocabulary, including all dialects. This database will be open to researchers who desire to work on any such system in the Pashto language for the greater good. We now have a desktop application, but we plan to migrate it to a standalone device so that it can be simply deployed.

8.1 Upgrading Dataset

We've created a prototype of our concept using five offensive terms and ten non-offensive words. However, in order for it to function in a real-time setting, the entire lexicon of Pashto speech must be covered. At start, we'll concentrate on learning the entire lexicon of the Yousafzai dialect. It could then be expanded to include other Pashto dialects.

8.2 Increasing Accuracy

Currently, our algorithm relies on 3000 15-word utterances from 200 different speakers. Our algorithm, which is based on CNN, has an accuracy of 85.3 percent. However, more data is required to improve its accuracy. By expanding the number of speakers per word, we will be able to add additional information.

8.3 Standalone Device

OLDUM's user interface is currently built in PyQt5, giving it a very user-friendly desktop program. Because we intend to use this software primarily for security organizations, we'll use a separate device to distribute it straight to their system. We'll concentrate on creating a proper standalone device with a Raspberry Pi 4 and an LCD display.

8.4 Mobile Application

Currently, OLDUM only has a desktop program which has been built in PyQt5. The model can easily be integrated with a mobile application, which then can be used on any mobile for detection purposes.

Bibliography

[1] Behr, V., Reding, A., Edwards, C., Gribbon, L.. 2013. Radicalisation in the Digital Era: The Use of the Internet in 15 Cases of Terrorism and Extremism. Santa Monica, CA: RAND.

[2] Rahman, Tariq. (1995). The Pashto language and identity-formation in Pakistan. Contemporary South Asia. 4. 151-170. 10.1080/09584939508719759.

[3] J. Ashraf, N. Iqbal, N. Khattak and A. Zaidi, "Speaker Independent Urdu Speech Recognition Using HMM", 2021.

[4] T. AlHanai, W. Hsu and J. Glass, "DEVELOPMENT OF THE MIT ASR SYSTEM FOR THE 2016 ARABIC MULTI-GENRE BROADCAST CHALLENGE", Groups.csail.mit.edu, 2021.

[5] Y. Kumar and N. Singh, "An automatic speech recognition system for spontaneous Punjabi speech corpus", International Journal of Speech Technology, vol. 20, no. 2, pp. 297-303, 2017.

[6] A. Arbab Waseem, N. Ahmad and H. Ali, "Pashto Spoken Digits Database for the Automatic Speech Recognition Research", 2012.

[7] Ahmed, Irfan & Ali, Hazrat & Ahmad, Nasir & Ahmad, Gulzar. (2012). The development of isolated words corpus of Pashto for the automatic speech recognition research. 139-143. 10.1109/ICRAI.2012.6413380.

[8] B. Zada and R. Ullah, "Pashto isolated digits recognition using deep convolutional neural network", Heliyon, vol. 6, no. 2, p. e03372, 2020.

[9] S. Khan, H. Ali, Z. Ullah, N. Minallah, S. Maqsood, A. Hafeez KNN and ANN-based recognition of handwritten Pashto letters using zoning features Mach. Learn., 9 (10) (2018)

[10] H.B. Chauhan, B.A. Tanawala Comparative study of MFCC and LPC algorithms for Gujrati isolated word recognition Int. J. Innovat. Res. Comput. Commun. Eng., 3 (2) (2015), pp. 822-826 View Record in Scopus

OLDUM

[11] M. A. Hossan, S. Memon and M. A. Gregory, "A novel approach for MFCC feature extraction," 2010 4th International Conference on Signal Processing and Communication Systems, 2010, pp. 1-5, doi: 10.1109/ICSPCS.2010.5709752.

[12] Hossan, Md & Memon, Sheeraz & Gregory, Mark. (2011). A novel approach for MFCC feature extraction. 1 - 5. 10.1109/ICSPCS.2010.5709752.

[13] Yujun Yang, Jianping Li and Yimei Yang, "The research of the fast SVM classifier method," 2015 12th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), 2015, pp. 121-124,doi:10.1109/ICCWAMTIP.2015.7493959.

[14] Evgeniou, Theodoros & Pontil, Massimiliano. (2001). Support Vector Machines: Theory and Applications. 2049. 249-257. 10.1007/3-540-44673-7_12.

[15] Tian, Yingjie & Shi, Yong & Liu, Xiaohui. (2012). Recent advances on support vector machines research. Technological and Economic Development of Economy. 18. 10.3846/20294913.2012.661205.

[16] Kaviani, Pouria & Dhotre, Sunita. (2017). Short Survey on Naive Bayes Algorithm. International Journal of Advance Research in Computer Science and Management. 04.

[17] Rish, Irina. (2001). An Empirical Study of the Naïve Bayes Classifier. IJCAI 2001 Work Empir Methods Artif Intell. 3.

[18] Dong T., Shang W., Zhu H. (2011) Naive Bayesian Classifier Based on the Improved Feature Weighting Algorithm. In: Shen G., Huang X. (eds) Advanced Research on Computer Science and Information Engineering. CSIE 2011. Communications in Computer and Information Science, vol 152. Springer, Berlin, Heidelberg.

OLDUM

[19] Sherstinsky, Alex. (2020). Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network. Physica D: Nonlinear Phenomena. 404. 132306. 10.1016/j.physd.2019.132306.

[20] N. M. Rezk, M. Purnaprajna, T. Nordström and Z. Ul-Abdin, "Recurrent Neural Networks: An Embedded Computing Perspective," in IEEE Access, vol. 8, pp. 57967-57996, 2020, doi: 10.1109/ACCESS.2020.2982416.

[21] Alex Sherstinsky, Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network.

[22] A. Rana, A. Singh Rawat, A. Bijalwan and H. Bahuguna, "Application of Multi Layer (Perceptron) Artificial Neural Network in the Diagnosis System: A Systematic Review," 2018 International Conference on Research in Intelligent and Computing in Engineering (RICE), 2018, pp. 1-6, doi: 10.1109/RICE.2018.8509069.

PLAGIARISM REPORT

ORIGINALITY REPORT

10%	6%	2%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to October University for Modern Sciences and Arts (MSA) Student Paper	1%
2	Submitted to University of Greenwich Student Paper	1%
3	www.coursehero.com Internet Source	1%
4	search.izlesene.com Internet Source	1%
5	Submitted to Higher Education Commission Pakistan Student Paper	1%
6	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	1%
7	Submitted to University of Hertfordshire Student Paper	<1%
8	Submitted to City University Student Paper	<1%

9	Submitted to University of Bradford Student Paper	<1 %
10	Submitted to Middlesex University Student Paper	<1 %
11	amslaurea.unibo.it Internet Source	<1 %
12	Submitted to CSU, San Jose State University Student Paper	<1 %
13	Submitted to Colorado Technical University Online Student Paper	<1 %
14	Submitted to Curtin University of Technology Student Paper	<1 %
15	Submitted to Central Washington UNiversity Student Paper	<1 %
16	Submitted to Universiti Tunku Abdul Rahman Student Paper	<1 %
17	faculty.ksu.edu.sa Internet Source	<1 %
18	spectrum.library.concordia.ca Internet Source	<1 %
19	K. Sreenivasa Rao, Manjunath K E. "Speech Recognition Using Articulatory and Excitation	<1 %

Source Features", Springer Science and Business Media LLC, 2017
Publication

20	Submitted to The Robert Gordon University Student Paper	<1 %
21	Submitted to Asia Pacific Instutute of Information Technology Student Paper	<1 %
22	Hussaini, Mohammed Abdullah, and Rabah W. Aldhaheri. "An Automatic Qari Recognition System", 2012 International Conference on Advanced Computer Science Applications and Technologies (ACSAT), 2012. Publication	<1 %
23	Submitted to Nanyang Technological University, Singapore Student Paper	<1 %
24	Submitted to Oxford Brookes University Student Paper	<1 %
25	Submitted to University of East London Student Paper	<1 %
26	Submitted to University of the West Indies Student Paper	<1 %
27	Dspace.lib.cranfield.ac.uk Internet Source	<1 %

29	SpringerBriefs in Electrical and Computer Engineering, 2015. Publication	<1 %
30	cije.up.pt Internet Source	<1 %
31	info.bioenabletech.com Internet Source	<1 %
32	www.codeproject.com Internet Source	<1 %
33	khirod-blog.blogspot.com Internet Source	<1 %
34	"Digital Heritage. Progress in Cultural Heritage: Documentation, Preservation, and Protection", Springer Science and Business Media LLC, 2016 Publication	<1 %
35	Submitted to University of Mauritius Student Paper	<1 %
36	docplayer.net Internet Source	<1 %
37	link.springer.com Internet Source	<1 %