

(SSTRUM)
SUSPICIOUS SPEECH TRACKING USING
MACHINE LEARNING



By

Bareedah Yousaf

Muhammad Ammad

Laiba Shehzeen

Maaz Ahmed


Submitted to the Faculty of Computer Science, Military College of Signals
National University of Sciences and Technology, Rawalpindi in partial fulfillment
for the requirements of a B.E Degree in Computer Software Engineering

JUNE 2021

CERTIFICATE OF CORRECTIONS & APPROVAL

It is certified that work contained in this thesis titled “*SSTRUM – Suspicious Speech Tracking Using Machine Learning*” is carried out by **Muhammad Ammad, Maaz Ahmed, Laiba Shehzeen and Bareedah Yousaf** under the supervision of **Asst. Prof. Mian Muhammad Waseem Iqbal and Asst. Prof. Dr. Shibli Nisar** for partial fulfillment of Degree of Bachelors of Computer Software Engineering, in Military College of Signals, National University of Sciences and Technology, Islamabad during the academic year 2020-2021 is correct and approved. The material that has been used from other sources it has been properly acknowledged / referred.

Approved by

Signature: 

Asst. Prof. Mian Muhammad Waseem Iqbal
(Supervisor)

Signature: 

Asst. Prof. Dr. Shibli Nisar
(Co-Supervisor)

Department of CSE, MCS

Dated: 20th June 2021

DECLARATION

It is declared that no part of SSTRUM thesis has been written, or taken from any other publication either in this institute or any other institute.

Plagiarism Certificate (Turnitin Report)

This Thesis document for the project SSTRUM has been evaluated for Plagiarism.

Turnitin report is attached at the end of the document.

Approved by

Signature: 

Asst. Prof. Mian Muhammad Waseem Iqbal
(Supervisor)

Signature: 

Asst. Prof. Dr. Shibli Nisar
(Co-Supervisor)

Department of CSE, MCS

Dated: 20th June 2021

ACKNOWLEDGEMENTS

We are thankful to our Creator Allah Subhana-Watala to have guided us throughout this project, in every thick and thin. Without His invaluable assistance and guidance, we would have been unable to accomplish anything. We owe a debt of gratitude to our families for their unwavering moral support, which has helped us become who we are. We are extremely grateful to our project supervisor Asst. Prof. Mian Muhammad Waseem Iqbal and project co-supervisor Asst. Prof. Dr. Shibli Nisar for their time, patience, and efforts they have spent on us. We would also like to thank our instructors, professors and all panelists in Military college of Signals (NUST) who taught us and helped us to complete our course work. We would also like to mention our friends and all of the people who have helped us with our research.

ABSTRACT

Terror activities are a bitter truth of our society. It has always been our governments first priority to avoid such activities by secretly spying the terrorists by different means. In the past decade, since ways of communication were not that strong so they were tracked manually. Now, with increasing advancement and automations, an increase in terror activities has been spotted, especially on special events where a large community is gathered. Definitely telephonic communication plays a major role in it. Security agencies are manually tracking every call to check if the speakers are trying to plan out any terror activities so that it could be avoided. However, this requires a large human resource since on average a specialized intelligence person could efficiently listen to telephonic conversations for 10 hours only. Government is now forced to even cut off telephonic signals on special gatherings. However, the problems rising due to technology should be dealt with technology.

SSTRUM – Suspicious Speech Tracking Using Machine Learning is our prototype solution to such technology related telephonic problems. This system will be capable of automating the process of speech tracking for Pashto language to avoid ill activities. With the use of machine learning algorithms our system will have accuracy of over 85% in the tracking of pre fed suspicious words, commonly used by trouble mongers.

A standalone device has been created for this purpose, which will be able to run our system at any place allowing the mobility, also serving any special occurrence efficiently. Our desktop application will be compatible with multiple operating systems with a user friendly interface which will help user track their audios with rapid response time.

TABLE OF CONTENTS

LIST OF TABLES	12
LIST OF ABBREVIATIONS	13
Chapter 1	14
Introduction	14
1.1 Overview	14
1.2 Problem Statement	14
1.3 Approach	14
1.4 Scope	14
1.5 Objectives	14
1.6 Deliverables	15
1.7 Justification for Selection of Topic	15
1.8 Overview of the Document	16
1.9 Document Conventions	16
1.10 Intended Audience	16
1.11 Thesis Outline	17
Chapter 2	18
Literature Review	18
2.1 What is speech?	18
2.2 Automatic Speech Recognition System	18
2.3 Definition of Natural Language Processing	19
2.4 How does an ASR System works?	19
2.5 Problems Faced in Making an ASR System	21
2.6 Previous work	22
Chapter 3	27
SSTRUM Methodology	27
Chapter 4	33
Software Requirement Specifications	33
4.1 Introduction	33
4.2 Overall Description	33
4.3 User Classes and Characteristics	33
4.4 Operating Environment	34

4.5 Design and implementation Constraints	34
4.6 User Documentation	35
4.8 External Interface Requirements	35
4.9 Communication Interfaces	36
4.10 System Features	36
4.11 Non-Functional Requirements	39
Chapter 5	41
Design and Development	41
5.1 Introduction	41
5.2 Work Breakdown Structure	41
5.3 Architectural Design	42
5.4 Decomposition Description	42
5.5 Use Cases	43
5.6 Sequence Diagrams	50
5.7 Data Design	54
5.7.1 Data Description	54
5.7.2 Data Dictionary	54
5.8 Component Design	55
5.9 Human Interface Design	61
5.9.1 Overview of User Interface	61
5.9.2 Screen Images	62
5.9.3 Screen Objects and Actions	67
Chapter 6	68
System Testing	68
6. Analysis and Evaluation	68
6.1. Introduction	68
6.2. Approach	68
6.3. Features to be tested	68
6.4. Pass/Fail Criteria	69
6.5. Testing tasks	69
6.6. Test Deliverables	69
6.7. Responsibilities:	69
6.8. Staffing and Training Needs:	69

6.9. Schedule	69
6.9.1. Important Dates	69
6.10. Risks and contingencies	70
6.10.1. Schedule Risk:	70
6.10.2. Operational Risks:	70
6.10.3. Technical risks:	70
6.10.4. Programmatic Risks:	70
6.11. Test Cases	70
6.11.1. Unit and Component level Testing	70
Chapter 7	76
Analysis	76
Chapter 8	78
Future Work	78
6.1 Increasing Accuracy	78
6.2 Enhancing Dataset	78
6.3 Standalone Device	78
Chapter 9	79
Conclusion	79
Chapter 10	80
Bibliography	80
Appendix A	83
Plagiarism Report	83

LIST OF FIGURES

Chapter 2:

Figure 2.1 Speech signal basics.....	18
Figure 2.2 General ASR system.....	19
Figure 2.3 MFCC extraction.....	20
Figure 2.4 Analog to digital.....	20
Figure 2.5 Spectrogram.....	21
Figure 2.6 Frames.....	21

Chapter 3:

Figure 3.1 Overall Block Diagram.....	27
Figure 3.2 MFCC extraction steps.....	28

Chapter 5

Figure 5.1 WBS.....	41
Figure 5.2 Block Diagram.....	42
Figure 5.3 Class diagram.....	42
Figure 5.4 - Use Case Diagram.....	44
Figure 5.5 – Login.....	50
Figure 5.6 Add User.....	51
Figure 5.7 View Users.....	51
Figure 5.8 Delete User.....	52
Figure 5.9 Add Recordings.....	52
Figure 5.10 Track Recordings.....	53
Figure 5.11 Logout.....	54
Figure 5.12 Login.....	55
Figure 5.13 Add User.....	56
Figure 5.14 View Users.....	57
Figure 5.15 Delete User.....	58

Figure 5.16 Add Recordings.....	59
Figure 5.17 Track Recording.....	60
Figure 5.18 Log Out.....	61
Figure 5.19 Home screen.....	62
Figure 5.20 Main Screen.....	62
Figure 5.21 Admin Panel.....	63
Figure 5.22 Login Page.....	63
Figure 5.23 Incorrect Password.....	64
Figure 5.24 Correct Password.....	64
Figure 5.25 Empty Password.....	65
Figure 5.26 Incorrect Password and username.....	65
Figure 5.27 Empty Username & Empty Password.....	66
Figure 5.28 Empty Username.....	66

LIST OF TABLES

Chapter 2:

Table 2.1 Research WER [3]	22
Table 2.2 Results [4]	23
Table 2.3 Phonetic files of Punjabi language.....	23
Table 3.4 Results [5]	24
Table 2.5 Results [7]	24
Table 2.6 Dataset [8]	25
Table 2.7 Research Analysis [8]	25

Chapter 4:

Table 4.1 Software Interfaces.....	36
------------------------------------	----

Chapter 6 :

Table 6.1 Admin/User Login.....	70
Table 6.2 Add New User.....	71
Table 6.3 View All Users.....	71
Table 6.4 Remove User.....	72
Table 6.5 Add Recordings.....	72
Table 6.6 Track Recordings.....	73
Table 6.7 Detect Suspicious Calls.....	73
Table 6.8 Alert Message.....	74
Table 6.9 Logout.....	75

Chapter 7:

Table 7.1. Accuracy of Algorithms.....	76
Table 7.2 CNN error rate.....	77

LIST OF ABBREVIATIONS

SSTRUM - Suspicious Speech Tracking Using Machine Learning

ASR – Automatic Speech Recognition

ADC – Analog to Digital Converter

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

1.1. Overview

SSTRUM is a complete machine learning based solution, which enables to automate the process of call tracking for Pashto language. It tracks out provided suspicious words out of recordings to make it less hectic for operators to track all calls manually.

1.2. Problem Statement

Monitoring of every voice call by the law enforcement authorities manually is a hectic job and this difficulty increases many folds when the terrorists use new codes for their operations [1]. Keeping up abreast of the new keywords and monitoring all calls manually is a gigantic problem faced by law enforcement agencies. Many efforts have been put forward by researchers to systematically digitize this process through machine learning techniques. Pashto language is still a problem for the agencies to cop with as Urdu and English based systems, all ready exists in the literature.

1.3. Approach

SSTRUM aims at developing a prototype of the system, using machine learning, which will be capable of detecting suspicious words in Pashto language, helping in automating the process of call tracing by law enforcement agencies and thus stopping any ill/terror activity.

1.4. Scope

The scope of our FYP is limited to isolated words only. We will be covering limited words of pashto language, which will be enough to train and deploy SSTRUM. In addition, initially, we will be training SSTRUM for isolated words and connected words only. Later on, after the completion of FYP prototype, the system can be upgraded to spontaneous speech.

1.5. Objectives

The main objective of this system is an efficient generation of suspicious speech tracker for Pashto language. 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 (SRS, SDS, Test Document, Final Report, and User manual). Students are also expected to develop extensive knowledge and technical skills in the following fields:

- Machine Learning
- Python Programming
- Creating Own Dataset
- Programming Raspberry pi

1.6 Deliverables

Sr No.	Tasks	Deliverables
1	Literature Review	Literature Survey
2	Requirements Specification	Software Requirements Specification document (SRS)
3	Detailed Design	Software Design Specification document (SDS)
4	Implementation	Project demonstration
5	Training	Deployment plan
6	Testing	Evaluation plan
7	Deployment	Complete desktop application with the necessary documentation

1.7. Justification for Selection of Topic

With increasing advancement and automations, an increase in terror activities has been spotted. In fact, Pakistan experienced numerous terrorist attacks in 2019. Despite the fact that Pashto is spoken a huge population, Automatic Speech Recognition for the Pashto is still not being explored to a extent as compared to other languages [2]. The majority of work in the past has only dealt with isolated digits.

1.8. Overview of the Document

This document shows the complete working process of our application SSTRUM. It starts with the literature review, which shows past work done in a similar field, requirement analysis of the system, system architecture which highlights the modules of the software and represents the system in the form of a component diagram, Use Case Diagram, Sequence Diagram and general design of the system. Then it will move on to discuss the detailed Description of all the components involved. Further, the dependencies of the system and its relationship with other products and the capacity of it to be reused will be discussed. In the end test cases and any future work, proposal has been presented.

1.9. Document Conventions

Headings are numbered in order of priority. Font used is Times New Roman. All the main headings are of size 16 and bold. All the second level sub-headings are of size 14 and bold. All the further sub-headings are of size 12 and bold. Where necessary references are provided in this document. However, where references are not provided, the meaning is self-explanatory.

1.10. Intended Audience

This document is intended for:

1. Developers: (Project Group)

To be certain that they are building up the correct venture that satisfies the necessities gave in this report.

2. Testers: (Project Group, Supervisor)

To have a definite rundown of the features and capacities that must react as indicated by prerequisites.

3. Users:

To be acquainted with the possibility of the task and how to utilize/react in disappointment circumstances and propose different highlights that would make it considerably progressively useful.

4. Documentation writers: (Project Group)

To recognize what features and how they need to clarify. What innovations are required, how the framework will react in every client's activity, what conceivable framework disappointments may occur, and what are the answers for each one of those disappointments, and so forth.

5. Project Supervisors: (Asst. Prof. Mian Muhammad Waseem Iqbal and Asst. Prof. Dr. Shibli Nisar)

This document will be used by the project supervisor to check whether all the requirements have been understood and, in the end, whether the requirements have been implemented properly and completely.

6. Project Evaluators: (CSE Dept. MCS)

To know the scope of the project and evaluate the project throughout the development of grading.

1.11 Thesis Outline

Thesis is divided in to following chapters:

Chapter 2: Basic definitions and Literature review

Chapter 3: Describes what ASR is and covers the functionalities of the proposed solution.

Chapter 4: This chapter covers the requirements specification of SSTRUM.

Chapter 5: This chapter discusses the Architecture of the system along with the data design.

Chapter 6: System Testing.

Chapter 7: Analysis of the system.

Chapter 8: Future work that is aimed to be produced.

Chapter 9: Conclusion

Chapter 10: Bibliography

Chapter 2

Literature Review

2.1 What is speech?

Speech is waves of changing air pressure, which are realized through excitation of vocal cords. It is generated when air is made to move by vocal cords radiating acoustic energy from vocal tract. These acoustic signals then cause listener's eardrum to move in and out depending upon the pressure fluctuations, thus transforming the acoustic energy into mechanical energy at eardrum. This mechanical energy, then reaches as neural energy to the listener's brain which is then processed as sound. Human speech frequency is $\sim 85\text{Hz} - 8\text{kHz}$ and hearing frequency is $\sim 50\text{Hz} - 20\text{kHz}$.



Figure 2.1 Speech signal basics

2.2 Automatic Speech Recognition System

Automatic Speech Recognition is a branch of Artificial Intelligence that enables the translation and recognition of spoken speech into text. It can also be described as a technology that let human beings talk to a computer interface like a normal conversation between human beings.

ASR Systems are serving in many different ways to automate different speech related processes. These are mainly divided into two different streams i.e. speaker dependent and speaker independent system. Speaker dependent systems are those in which system will 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 in case of speaker independent speech recognition system, the system does not rely on vocal training and responds to every speaker.

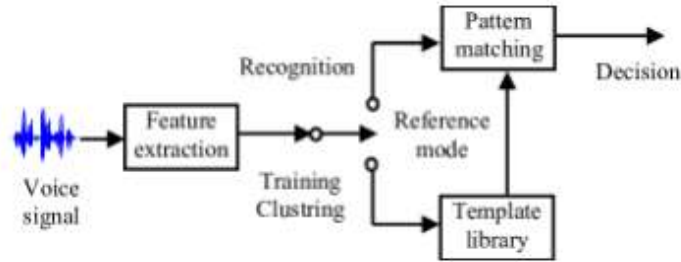


Figure 2.2 General ASR system

2.3 Definition of Natural Language Processing

Natural Language Processing or NLP is the most advanced version of the ASR systems. It allows human to interact with computer through their voices and text. We humans interact with each through both these forms i.e. written text and voices. Therefore, In order to make our lives easier, we need to invent ways to make our computers understand us in both these forms. In addition, NLP is the solution to all of this.

Study of NLP has been around since 5 decades now and it is still growing. We can clearly see the results in form of our smartphones responding to our voices as Siri or Bixbi. Accuracy of around 96 to 99% has been achieved in different NLP programs. It is programmed on a very large vocabulary and as it is almost impractical to scan the whole vocabulary hence it is designed to mark the tagged words only and respond to them.

2.4 How does an ASR System works?

As a first step noise is removed from the audio data and then features are extracted from the data using Mel-Frequency Cepstral Coefficient (MFCC). To extract features, audio needs to be passed through few steps as represented in the block diagram below.

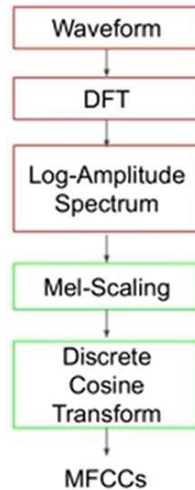


Figure 2.3 MFCC extraction

The speech signals are firstly converted from analogue to digital signals. It is because computer cannot process on analogue data. This is done by Analogue to Digital Converter (ADC) in three steps i.e. sampling, quantization and encoding.

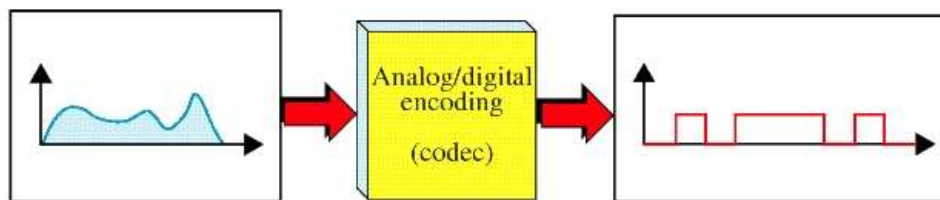


Figure 2.4 Analog to digital

Once the data has been converted into digital format, we then apply Fast Fourier Transform (FFT) to convert the graph in spectrogram. This spectrogram displays frequency on the vertical axis, time on the horizontal axis and the color shows the intensity that actually makes the sound. The lighter the color the more energy is used.

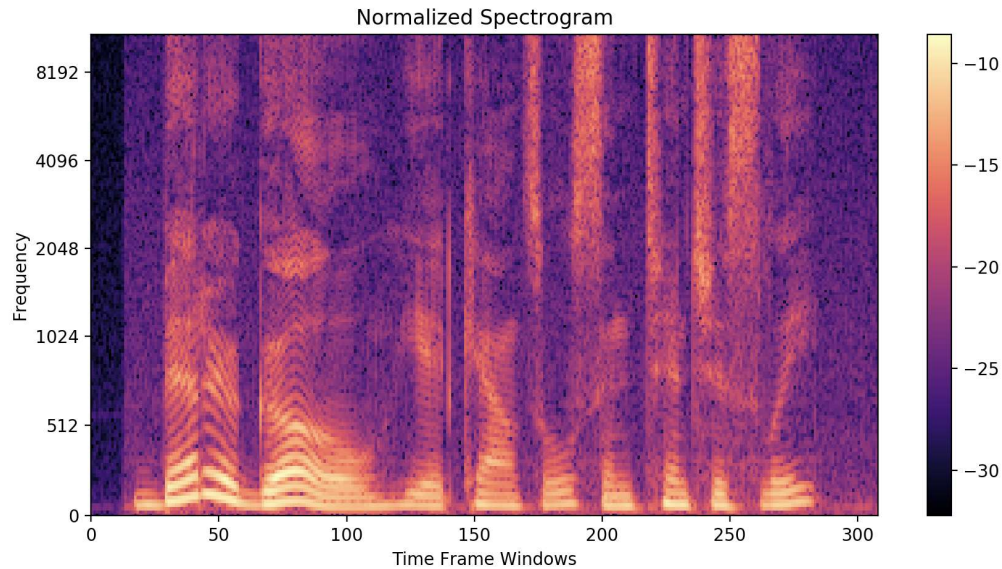


Figure 2.5 Spectrogram

After it is done the features of data can be extracted. Usually first 12 to 13 coefficient are considered most important because they contain most of the important information like formants and spectral envelopes etc. In total there are 39 coefficients per frame.

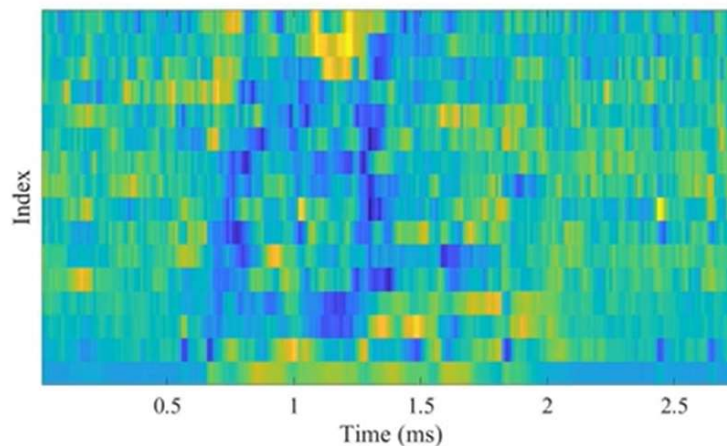


Figure 2.6 Frames

After these features are extracted, the required algorithm is applied depending on the system.

2.5 Problems Faced in Making an ASR System

With world moving towards this technology for their ease, following are the major problems which are faced in deploying such systems in real life everywhere. The biggest of all problems in noise.

Noise is everywhere around us and it disrupts the sound waves making it difficult for the computer to understand and differentiate between different words.

Accent is another huge barrier in enhancing the efficiency of ASR systems. Every language has multiple accents and if a word is pronounced in many different ways, then the phonemes for each accent will vary making it difficult for an ASR system to process.

Moreover, echo and similar sounds of words is also a problem but apart from all that the speed by which this technology is enhancing, all big companies are working on making their own ASR systems to enable automation of all tasks.

2.6 Previous work

This section covers the previous work done on ASR systems and on Pashto dataset formation. In the recent decade, a huge amount of research was carried on local languages including Urdu, Punjabi, Hindi, and Arabic.

1. Speaker Independent Urdu Speech Recognition using HMM

Javed Ashraf et al [3], presented an ASR system for limited vocabulary that was speaker-independent. This system works for isolated words only. The system is based on Sphinx4 and HMM having a word error rate of 10%.

From this research work, it was concluded that Sphinx-4 framework could efficiently work for small to medium sized vocabulary. The acoustic model used for this system was developed entirely by the authors of this project.

Speaker	WER (%)			
	Test1	Test2	Test3	Mean WER
Speaker1	10	10	10	10.00
Speaker2	10	10	20	13.33
Speaker3	10	0	10	6.66
Speaker4	20	10	10	13.33
Speaker5	10	10	10	10.00
<i>Mean</i>				10.66

Table 2.1 Research WER [3]

2. Development of the MIT ASR system for the 2016 Arabic multi-genre broadcast challenge

Tuka AlHanai et al [4], described an Arabic Automatic Speech Recognition system developed on a 1,200-hour speech corpus. A range of Deep Neural Network (DNN) topologies was modelled and it was noted that the best performance with respect to time came from a sequence discriminatively trained G-LSTM neural network. CNN model was found good at feature extraction representation and reducing variance in frequency domain. Thus, better results could be concluded from it if it was used within a hybrid-like DNN topology. The best overall Word Error Rate was 18.3% ($p < 0.001$). The significance of the results (8/13 results with $p < 0.05$ compared to next increase in WER) highlights that each incremental improvement in WER introduced by a different network topology is a significant increase, even if it is a difference of only 0.3% absolute.

Model	Topology	Features	Alignments	WER (%) $p < (\text{prev}/\text{base})$	WER (%) 4gram $p < (\text{prev}/\text{base})$
GMM-HMM	-	MFCC+LDA+MLLT+FMLLR	-	40.3 (-/-)	-
DNN CE	5x1024	30 Fbank + Pitch	GMM	29.7 (0.001/0.001)	28.1 (0.001/0.001)
CNN	4x2000	80 Fbank + Pitch	GMM	29.5 (0.472/0.001)	28.1 (0.734/0.001)
TDNN	6x3000	80 Fbank + Pitch	GMM	27.1 (0.001/0.001)	25.8 (0.001/0.001)
DNN MPE	5x1024	30 Fbank + Pitch	CE	25.6 (0.001/0.001)	24.7 (0.001/0.001)
Chain TDNN	7x625	80 Fbank + Pitch	GMM	23.6 (0.001/0.001)	23.4 (0.001/0.001)
LSTM	3x1024	80 Fbank + Pitch	CE	23.6 (0.936/0.001)	22.7 (0.001/0.001)
H-LSTM 3L	3x1024	80 Fbank + Pitch	CE	23.3 (0.027/0.001)	22.6 (0.250/0.001)
H-LSTM 5L	5x1024	80 Fbank + Pitch	CE	23.1 (0.055/0.001)	22.4 (0.184/0.001)
G-LSTM 3L	3x1024	80 Fbank + Pitch	CE	22.4 (0.001/0.001)	21.7 (0.001/0.001)
G-LSTM 5L	5x1024	80 Fbank + Pitch	CE	22.2 (0.110/0.001)	21.5 (0.070/0.001)
G-LSTM 3L sMBR	3x1024	80 Fbank + Pitch	CE	20.4 (0.001/0.001)	19.5 (0.001/0.001)
G-LSTM 5L sMBR	5x1024	80 Fbank + Pitch	CE	20.1 (0.009/0.001)	19.2 (0.034/0.001)
Top 2 Combined	G-LSTM sMBR (3L+ 5L)	80 Fbank + Pitch	CE	-	18.3 (0.001/0.001)

Table 2.2 Results [4]

3. An automatic speech recognition system for spontaneous Punjabi speech corpus

An automatic speech recognition system for spontaneous Punjabi speech corpus was developed by Yogesh Kumar et al [5]. The system was capable to recognize the spontaneous Punjabi live speech. The user interfaces for the Punjabi live speech system were created by using java programming. The system was trained with 6012 Punjabi words and 1433 Punjabi sentences. The main objective of this research work was to reduce the %error in the speech model.

ਸ	ਭ	ਤ	ਰ	ਨ	ਹ	ਲ	ਕ
ਪ	ਜ	ਦ	ਠ	ਅੰ	ਆ	ਪਿੰ	ਕਿੰ
ਮੈ	ਰਾ	ਓ	ਅ	ਬ	ਮ	ਨਾ	ਯ
ਗ	ਸ	ਸਿੰ	ਖਿ	ਕੰ	ਪਿ	ਊ	ਇੰ
ਨਿੰ	ਐ	ਈ	ਵਿ	ਚ	ਟ	ਕਿੰ	ਪੁ
ਸੀ	ਖ	ਊ	ਘ	ਕੁ.	ਨੂੰ	ਬ	ਫੁ
ਦਿ	ਚਿ	ਰਾਂ	ਵ	ਧੰ	ੌ	ੇ	ਾਂ
ਂ	ੈ	ੀ	ੇ	ਾਂ	ੌ	ੂ	!

Table 2.3 Phonetic files of Punjabi language

This accuracy of this system was tested during a Punjabi interview. As mentioned in table below, 1227 Punjabi words and 461 Punjabi sentences were uttered. The performance measured in terms of recognition accuracy was 90.8% for Punjabi sentences and 93.79% for Punjabi words.

Punjabi Interview Speech Corpus ↓	Correct	Error	Accuracy (Correct Percentage)
Total Number of Punjabi Sentences = 461	455	6	98.6 %
Total number of Punjabi words = 1227	1213	14	98.8 %

Table 2.4 Results [5]

4. Pashto Spoken Digits Database for the Automatic Speech Recognition Research

Recently a little work was also done on the Pashto language but it was restricted to isolated digits only. Arbab Waseem et al [6], presented the development of the first Pashto isolated Spoken Digits database for the automatic speech recognition research consisting of digits from zero (sefer) to hundred (sul) uttered by sixty speakers, 30 male, and 30 female. The speakers in attendance ranged in age from 18 to 60 years old. The recordings were made with a Sony PCM-M 10 Linear Recorder

in a noise-free environment. The feature vector was Mel Frequency Cepstral Coefficients (MFCC), and the classification was done with a Linear Discriminant Analysis (LDA) based classifier. The performance measured in terms of recognition accuracy was found to be 67%.

5. The Development of Isolated Words Pashto Automatic Speech Recognition System

Some other relevant work was done by Irfan Ahmed et al [7] in the Pashto language with a medium-sized corpus. The dataset consists of 50 utterances of 161 different isolated most commonly used words of the Pashto language, names of the days of the week, and digits from 0 to 25. The dataset contains a recording from both genders i.e. 28 male and 22 female speakers having ages ranging from 16 to 40. The sampling frequency was 44100 Hz. The word error rate was calculated to be 33.33%.

Word number	Percentage Error
1	0
2	33.33
3	0
4	60
5	33.33
6	33.33
7	50
8	33.33
9	33.33
10	33.33

Table 2.5 Results [7]

6. Pashto isolated digits recognition using deep convolutional neural network

Following the above work, Pashto isolated digits' recognition using deep convolutional neural network was presented by Bakht Zada et al [8], showing better performance as compared to previous similar works. The vocabulary consisted of 50 utterances for each Pashto digit from 0 to 9. Twenty MFCC features were extracted for each isolated digit and fed as input to CNN. The total average of 84.17% accuracy was achieved from testing which clearly shows better accuracy than previous systems. Some more work done for Pashto is shown in [9] [10].

Data	Number of Speakers	Accuracy
Training	25 (both male and female)	90.14%
Testing	25 (both male and female)	84.17%

Table 2.6 Dataset [8]

Research.P#	Language	Algo	Accuracy	WER
1	Urdu	Sphinx4	90%	10%
2	Arabic	DNN	81.7%	18.3%
3	Punjabi	CNN	90.8%	9.2%
4	Pashto	LDA	67%	33%
5	Pashto	DNN	66.67%	33.33%
6	Pashto	CNN	84.17%	15.83%

Table 2.7 Research Analysis [8]

Chapter 3

SSTRUM Methodology

The system was designed in a way that the whole process is as smooth and precise as possible, and for this reason, we have mentioned the whole methodology in figure 3.1. Our speech signal after the initial processing is given to the system as an input to be further inspected. Firstly, the feature extraction process is carried from the speech signal which will be explained below. For the recognition, we used the CNN model for our system as a priority. This was used to train our model. We also used different other models such as SVM classifier, Naïve Bayes Algorithm, Random Forest classification model, etc. to check for accuracy on these models. We will also be explaining the details of these models in the paper briefly. The training was carried on almost 80 % of our data and then accuracy was determined for respective Machine Learning Algorithms. The output of the system was then shown in the form of suspicious or non-suspicious word labels in our desktop application interface.

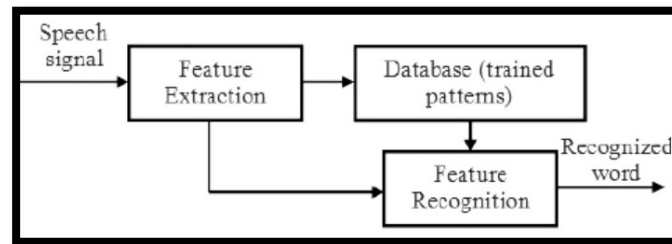


Figure 3.1 Overall Block Diagram

3.1 MFCC Feature Extraction

The speech signals have some unnecessary data which we label as noise. This audio signal is required to be filtered to eradicate such noise before the step of extracting the features. Feature extraction [11] is done to evaluate the speech signal and illustrate the signal in a certain number of components of this signal which are useful. This also removes the unimportant parts of speech.

In sound processing, we commonly use the representation of the short-term power spectrum of a sound in the shape of mel-frequency cepstrum coefficient (MFCC). Mel-frequency cepstral coefficients (MFCCs) are coefficients that collectively make up an MFC. These are derived from

a nonlinear spectrum of the audio clip. Our system has also used the MFCC feature extraction technique.

The technique of MFCC feature extraction is based on different process that are labelled as pre-emphasis, Frame blocking, windowing, taking FFT, then equating the frequencies on a Mel scale, and at the end inverse DCT is applied. The block diagram of the process in figure 3.2 gives us an overview of the whole process, which will be further explained below.

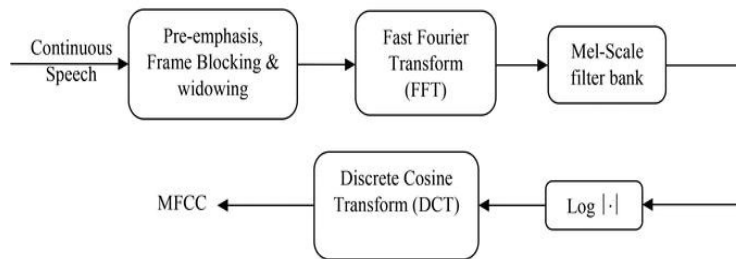


Figure 3.2 MFCC extraction steps

A. Pre-emphasis: The term "pre-emphasis" refers to the process of filtering a signal so that it focuses on higher frequencies. It's most commonly used to balance the spectrum of voiced sounds with a steep roll off in the high frequency range. 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 slowly time varying signal. The speech signal has to be examined over a short period of time for stable acoustic characteristics. This is why speech analysis is carried on short segments across which the speech is stationary. These short term measurements are carried out in windows of 20 ms or 10 ms. This enables individual speech sounds with temporal characteristics to be tracked and this window is sufficient to provide good quality spectral resolution of sound. Hamming windows are used normally to enhance the harmonics and reduce the edge effect during Fourier transform.

C. Fourier transform: The windowed frames are the converted to magnitude domain (time to frequency) by applying Fourier transform.

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

D. Mel spectrum: The Mel spectrum is the next step which is taken by applying a Mel filter bank on Fourier transformed signal. This is unit of measure which is based on human ears perceived frequency. The Mel scale is approximately a linear frequency spacing below 1 kHz and a logarithmic spacing above 1 kHz.

E. Discrete cosine transform (DCT): Discrete cosine transform (DCT) converts the log Mel spectrum into the time domain. The output of this step is the final MFCC features that were required. The distinct number of different coefficients obtained by applying DCT are deemed as acoustic vectors. 13 different coefficients were extracted for our system [12].

3.2 Machine Learning Algorithms

The main algorithm that we opted for our system was CNN but for comparative study, we also analyzed behavior of other machine learning algorithms. Before going into analysis of the algorithms, we will be discussing basics here for each one.

3.2.1. Convolutional Neural Network

Convolutional neural networks is a machine learning-based model which works on a mathematical operation of convolution. Here we use convolution rather than the general multiplication of matrices in one of the layers. This neural network has different layers labeled as the input, output, and hidden layer.

The middle layer is the one that is the hidden one because the activation function and the convolution mask the inputs and outputs of this layer. These hidden layers undertake the process of convolution. Here dot product is taken between the inputs of the layer i.e. in our case the MFCC features and the convolutional kernel. This kernel is then slid across the input matrix and generates a feature map which is deemed as the input for the next layer. Further, the processes of pooling and normalization layer etc. are performed.

Along with the convolutional layer, there is also a pooling layer that reduces the data by combining the data clusters of the output of one layer to a distinct cluster for the upcoming layer. Max pooling

and average pooling are the two most common types of pooling where the maximum value of each cluster and the average value is taken respectively.

The next step is dropout where a neural network is prevented from overfitting and for this purpose, different probability values are tested for better accuracy. We will be discussing the results for our dataset while the convolutional neural network was applied to it in the next sections.

3.2.2 SVM

A support vector machine (SVM) [13] is a supervised machine learning model. SVM is commonly utilized to classify two entities. It is basically used on labeled data. It is more of a binary classifier that can guess if a vector belongs to class 1 or class 2. In our case, the data was to be classified into suspicious or non-suspicious. Here hyperplane concept is used to classify data where we have to determine if our p dimensional plane data can be classified in a $p-1$ dimensional hyperplane. One of the best hyperplanes can be the one where we have to measure the one that represents the largest separation between the two groups. This one is called the maximum margin hyperplane.

In some cases, it is seen that the regional data is in finite-dimensional space but the sets that are used to discriminate are not in a state where they can be linearly separable in space. Here this original data is mapped into a high dimensional space where we can easily separate it. A kernel function is defined which is used to ensure that the dot product of input vectors can be easily computed in terms of the original space variables. Here, a sum of kernels can be used to measure the distance of each test point to the data points to be defined.

SVM is somewhat more effective than other methods in the criteria upon which it learns the data. Our main aim is to reduce the number of misclassifications. SVM can work on a comparatively smaller dataset with higher accuracy depending upon the quality of data. One of the biggest advantages is that it gives a guaranteed unique solution and can even work well in noisy environments [14] [15].

3.2.3 Naïve Bayesian algorithm

It is an algorithm [16] which is initiated from the Bayes theorem, which works on conditional probabilities i.e. occurrence of one event based on the occurrence of the other [17].

$$P(B|A) = \frac{P(A \text{ and } B)}{P(A)}$$

In this algorithm, it is assumed that each attribute makes an equal and independent impact on the whole outcome [18].

3.2.4 Recurrent Neural Networks

Recurrent neural networks (RNN) [19], defined as a machine learning algorithm that is used for sequential data. This is an algorithm, which is specific in its application of internal memory that helps it to remember the inputs. It has proven to be vital in deep learning and even Google voice search and Siri of Apple are using it.

In this algorithm, the information runs through a loop while making a decision. It not only considers the current input but also the knowledge of previous inputs. It records the characters in its internal memory and copies the output to be looped back into the network. It in simple terms adds the past values to the current ones [20]. Hence, it has two inputs which are the current ones and the ones in the recent past [21].

3.2.5 Multilayer Perceptron

A multilayer perceptron (MLP) [22] is a type of feedforward artificial neural network. It has at least three layers of nodes, which are labeled as the input, output, and the hidden layer. Every node uses a nonlinear activation function except or input layer [23].

It can be helpful in determining the data that is not separable linearly. The major use cases of MLP are recognition, prediction, approximation, and pattern classification .

3.2.6 Random Forest Classification

This algorithm [24] works by constructing multiple decision trees at a certain training period and then giving that output to the class that is meant for classification and prediction of the individual trees. It is a supervised learning algorithm.

It works on a bagging method, which means that the more the combination of models, the better the overall result will be. It simply constructs multiple decision trees and combines them together for better accuracy in the result. It can be used for both regression and classification problems.

SSTRUM

While growing the trees, it adds more randomness to the model. Instead of looking for a specific feature when splitting a node, it looks for the best feature out of a list of options. This results in a more diverse result, and thus a better overall result.

Chapter 4

Software Requirement Specifications

4.1 Introduction

This chapter provides an overview of the entire Software Requirement Specification document with the purpose, scope, functional and non-functional requirements of our system. The aim of this document is to present a detailed description of the project SSTRUM which aims to develop a suspicious speech tracking device to automate the process of tracking calls. The detailed requirements of SSTRUM is provided in this document.

4.1.1 Purpose

The purpose of this chapter is to present a detailed description of the software that does Suspicious Speech Tracking in Pashto language, helping in automating the process of call tracing by agencies, to avoid any terror activity. This is the basis of the guideline for the entire software development. It will also be useful for the clients to ensure all requirements and specifications.

4.2 Overall Description

4.2.1 Product Functions

The main features of SSTRUM in this domain are highlighted below:

- The system records suspicious words.
- Users will provide call audios as input to the system.
- It will monitor calls based on suspicious words stored in the dataset.
- SSTRUM will determine if the call audio contains any suspicious words.
- All the process will be shown on a dedicated desktop application.

4.3 User Classes and Characteristics

The following section describes the types of users of SSTRUM.

4.3.1 Security Agencies

Security agencies will have the access to the system who will be able to even extend the dataset according to their needs and resources once the prototype is in their hands.

4.3.2 Corporate sector

Different companies will be able to keep their communications secure while protecting their privacy by using relevant datasets.

4.4 Operating Environment

4.4.1 Hardware

SSTRUM will have following hardware specifications:

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

4.4.2 Software

SSTRUM will have following Software specifications:

- Linux/Raspbian
- Python IDE

4.5 Design and implementation Constraints

- It will be able to process one call at a time.
- The dataset will be limited to selected words and will be trained according to those words, although it will have an option to expand the dataset according to the needs.
- The system will need high processing power, so we will have to take care of those specifications.

4.6 User Documentation

The users will be given a user manual in which they will be given instructions on how to operate SSTRUM along with the limitations that need to be taken into consideration. Users will also have access to a project report that illustrates the software's features, functionality, and procedures.

4.7 Assumptions and Dependencies

- Constant power supply
- Users must know the limitations in the dataset and that the system will have to be trained for a separate dataset in case they want to make it adaptable to their organization.
- The accuracy of the system will have to be taken into account.

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 suspicious 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	For training purposes of our system, we have chosen Windows 10 OS for its best support and user-friendliness. And for the end user interface, we will be using Raspbian OS.
Database	The user information will be saved in a database, and a login/signup interface will appear every time for users to enter their credentials.
Python	We have chosen Python as our main programming language as training the machine and all techniques is relatively simpler and more efficient.

Table 4.1 Software Interfaces

4.9 Communication Interfaces

Our system runs with Python in the background processing (monitoring for suspicious words) every audio file sent or being listened to real time and a UI based on the Raspbian OS giving real-time feedback. A login/signup feature has also been added for securing the entire process and the system.

4.10 System Features

SSTRUM will be providing following system features:

1. Signup

- **Description and Priority**

User will first signup/register to SSTRUM 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 SSTRUM.

- **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 SSTRUM for further processing on audio files and detecting suspicious calls. 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 SSTRUM.

- **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**

SSTRUM 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 calls 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 suspicious or not suspicious

- **Description and Priority**

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

- **Stimulus/Response Sequences**

1. The user will upload the audio files.

2. The system will match audio words with already stored suspicious 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 suspicious and nonsuspicious words.

5. Alert the user for suspicious calls

- **Description and Priority**

SSTRUM will immediately alert the user if it detects any suspicious words in the audio file or call.

It is of very high priority because the user needs to be alerted of the suspicious calls to act immediately to stop the ill/terror activities.

- **Stimulus/Response Sequences**

1. The user will upload audio files.

2. The system will match the audio words with the suspicious words in the dataset.

3. Upon detecting suspicious words, the user will be alerted on the screen for suspicious call.

- **Functional Requirements**

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

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

6. Update Dataset/Suspicious words list

- **Description and Priority**

User will be able to update SSTRUM for new suspicious 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 suspicious words to the dataset.
2. The user will then train the system to detect new suspicious words.
3. The user will test the system to detect the new suspicious words from the audio files.

- **Functional Requirements**

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

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

7. Sign out

- **Description and Priority**

User will logout of SSTRUM 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

- The response time of the software must not be too long,
- 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 Safety Requirements

- Application shall handle any user's information safely.
- Users must have to register using original information so that if any mishap occurs service shall provide him as much support as possible.
- User credentials and private info shall not be shared with the rest of the users

4.11.3 Security Requirements

- Only authorized users can access the system to prevent any unwanted changes in it.
- System shall not be accessed by unauthorized person.

4.11.4 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 suspicious words.
- **Reusability:** The components of the system shall be written in a way that they are easy to reuse.
- **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.
- **Usability:** The software should be easy to use for the users.

Chapter 5

Design and Development

5.1 Introduction

This chapter mainly covers the work breakdown structure of our system into small deliverables along with the system architecture and system design based on all our functional requirements described in the above chapter.

5.2 Work Breakdown Structure

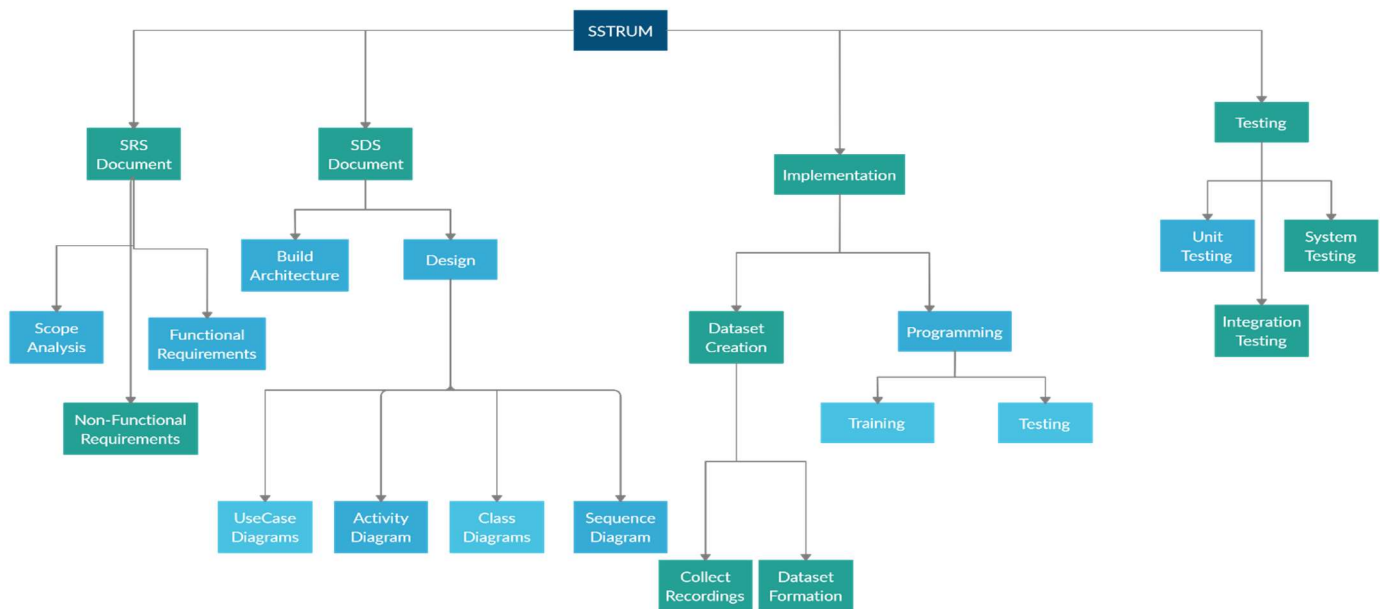


Figure 5.1 WBS

5.3 Architectural Design

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

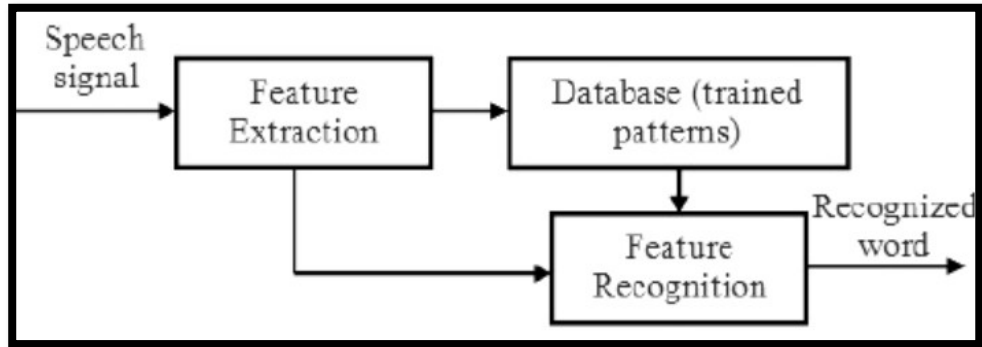


Figure 5.2 Block Diagram

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.

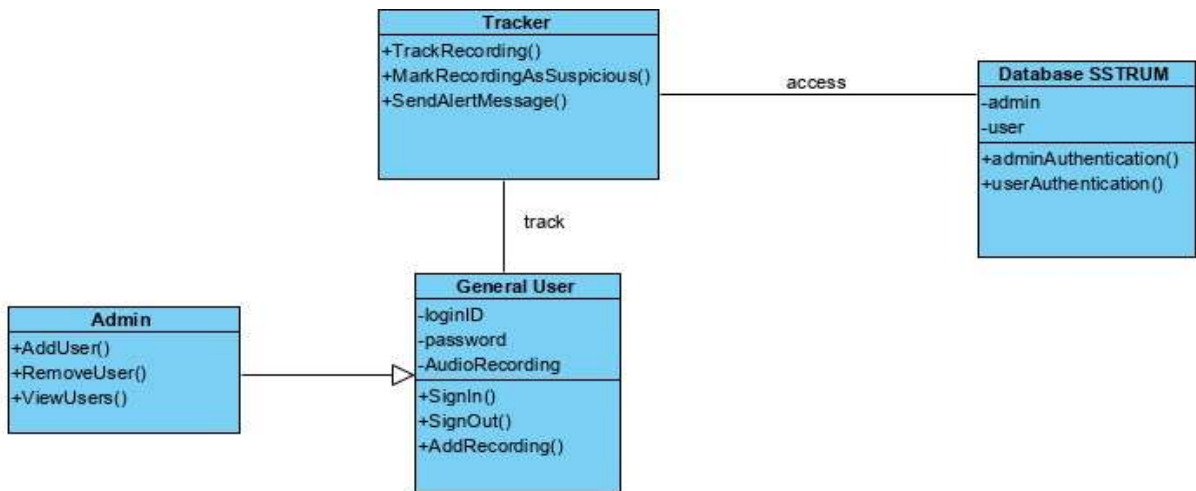


Figure 5.3 Class diagram

5.5 Use Cases

A use case is a method for identifying, clarifying, and organizing system requirements in the context of system analysis. A use case is a collection of possible interactions between systems and users in a specific environment, all of which are related to a specific goal.

The various user classes identified the following use cases and primary actors for SSTRUM:

Actors	Use Cases
Admin	<ul style="list-style-type: none">• Login• Manage Users• Add Recordings• Start Tracking• Log out
Users	<ul style="list-style-type: none">• 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

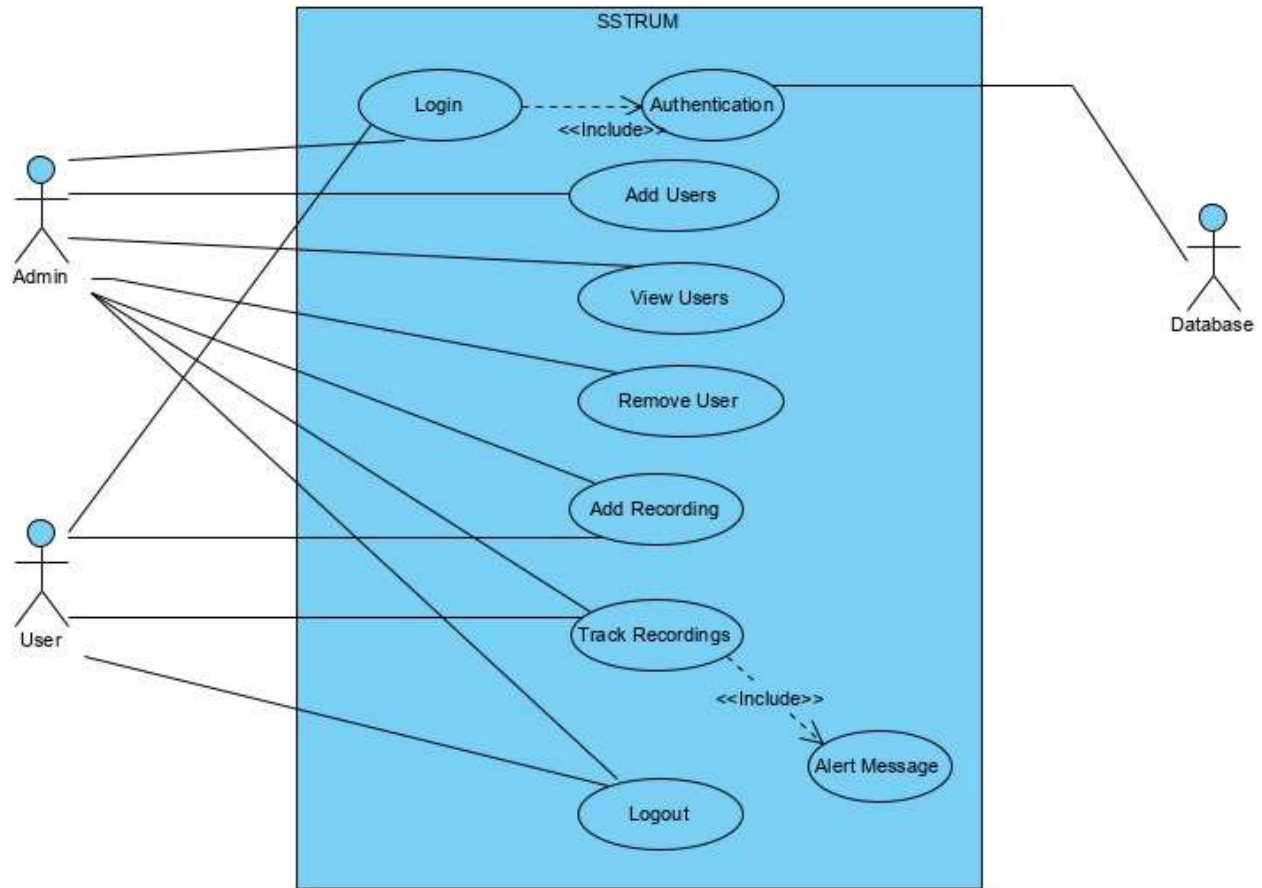


Figure 5.4 - Use Case Diagram

5.3.2 Use Cases Description

1. Login

Use Case ID:	1		
Use Case Name:	Login		
Actors:	Admin, Users		
Created by:	Bareedah	Last Updated by:	Bareedah
Date Created:	24/01/2021	Date Last Updated:	24/01/2021
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 the user enters invalid user name and password, the system state remains unchanged and it tells user to re-enter his credentials.
Normal Flow (Primary Scenario):	<ol style="list-style-type: none"> 1. The system requests that the actor enter his/her private credentials i.e. user name and password. 2. The actor enters his/her user name and password. 3. The system verifies the entered user name and the password and logs the actor into the system.
Alternative Flows:	If in the Basic Flow, the actor leaves the fields empty or enters an invalid user name and/or password the system displays an error message. The actor can choose to either return to the beginning of the Basic Flow or cancel the login, and at this point the use case ends.

Table 5.1 Login use case description

2. Add Users

Use Case ID:	2		
Use Case Name:	Add Users		
Actors:	Admin		
Created by:	Laiba	Last Updated by:	Laiba
Date Created:	24/01/2021	Date Last Updated:	24/01/2021
Description:	Admin has to log in to the system and add new users to allow them access to SSTRUM.		
Preconditions:	Admin has to log in.		
Post conditions:	New users should be updated in the database.		

Normal Flow (Primary Scenario):	<ol style="list-style-type: none"> 1. The admin will login and switch to Admin panel. 2. The Admin will add new users. 3. Changes will be updated in the database.
Alternative Flows:	<ol style="list-style-type: none"> 1. An error is encountered during the modification of database. 2. Proper functionality of the database will be checked.

Table 5.2. Add user use case description

3. Remove Users.

Use Case ID:	3		
Use Case Name:	Remove Users		
Actors:	Admin		
Created by:	Laiba	Last Updated by:	Laiba
Date Created:	24/01/2021	Date Last Updated:	24/01/2021
Description:	Admin has to log in to the system and remove users to disallow access to SSTRUM.		
Preconditions:	Admin has to log in.		
Post conditions:	User should be removed from the database.		
Normal Flow (Primary Scenario):	<ol style="list-style-type: none"> 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:	<ol style="list-style-type: none"> 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:	Bareedah	Last Updated by:	Bareedah
Date Created:	24/01/2021	Date Last Updated:	24/01/2021
Description:	Admin has to log in to the system and view the registered users.		
Preconditions:	Admin has to log in.		
Post conditions:	List of registered users should be displayed.		
Normal Flow (Primary Scenario):	<ol style="list-style-type: none"> 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:	<ol style="list-style-type: none"> 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:	Ammad	Last Updated by:	Ammad
Date Created:	24/01/2021	Date Last Updated:	24/01/2021

Description:	When Users login to their profile, they will be provided access to SSTRUM. They can add new recordings to track out suspicious words from them.
Preconditions:	User has to log in.
Post conditions:	Audio name will be displayed on the screen on successful upload.
Normal Flow (Primary Scenario):	<ol style="list-style-type: none"> 1. The user will login to the profile. 2. User can add recordings.
Alternative Flows:	<ol style="list-style-type: none"> 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:	SSTRUM		
Created by:	Ammad	Last Updated by:	Ammad
Date Created:	24/01/2021	Date Last Updated:	24/01/2021
Description:	After Uploading audio, user has to click Start Tracking button for SSTRUM to analyze the audio.		
Preconditions:	User has to log in and upload an audio.		
Post conditions:	SSTRUM will analyze the audio.		
Normal Flow (Primary Scenario):	<ol style="list-style-type: none"> 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:	Maaz	Last Updated by:	Maaz
Date Created:	24/01/2021	Date Last Updated:	24/01/2021
Description:	Actor attempts to log out of the system.		
Preconditions:	Actor has to be logged in first.		
Post conditions:	The Actor will be logged out and sent to the login page.		
Normal Flow (Primary Scenario):	<ol style="list-style-type: none"> 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

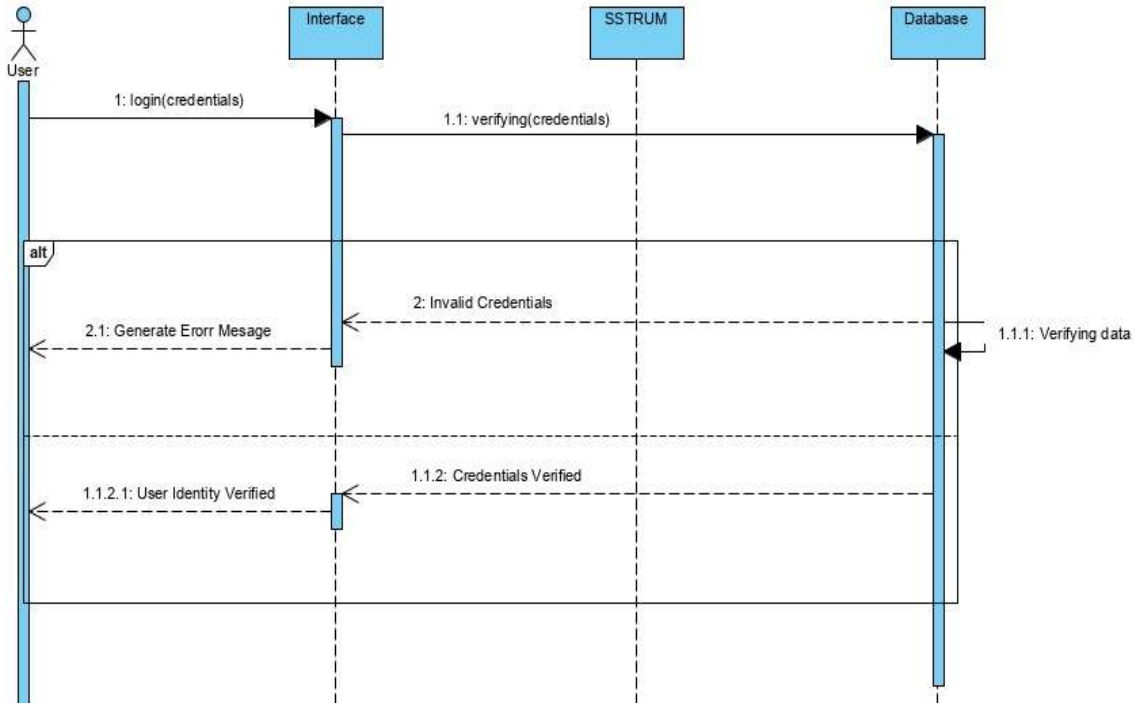


Figure 5.5 – Login

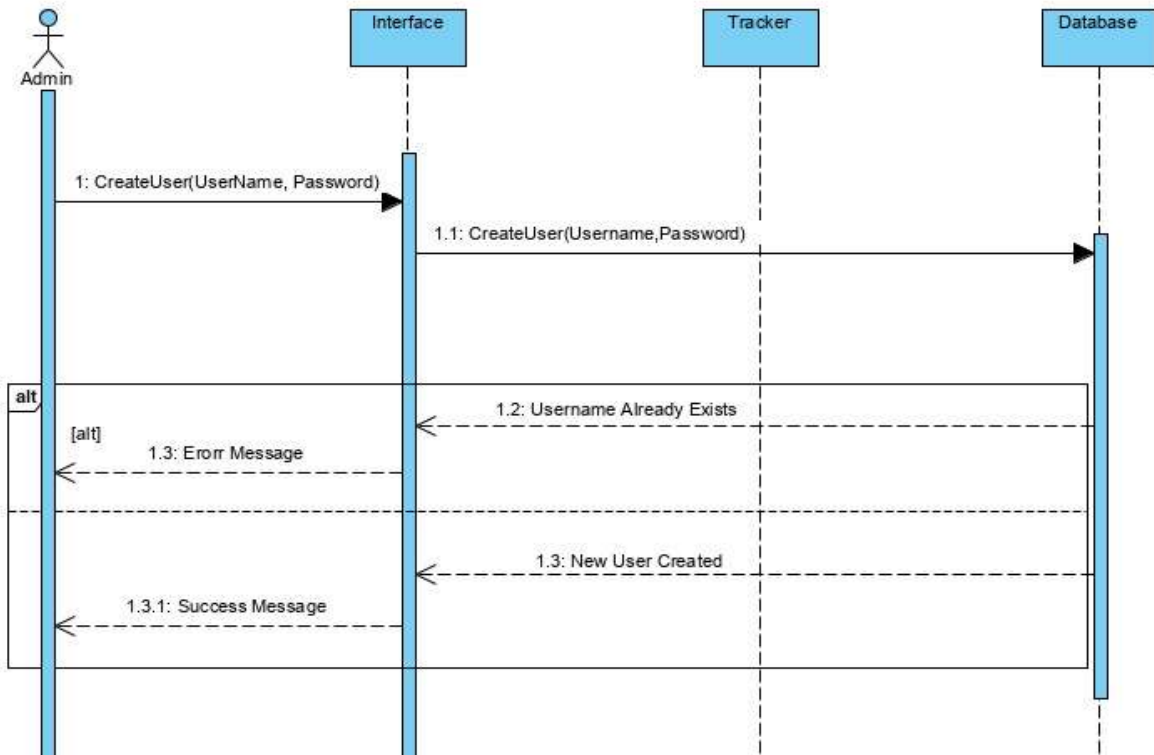


Figure 5.6 Add User

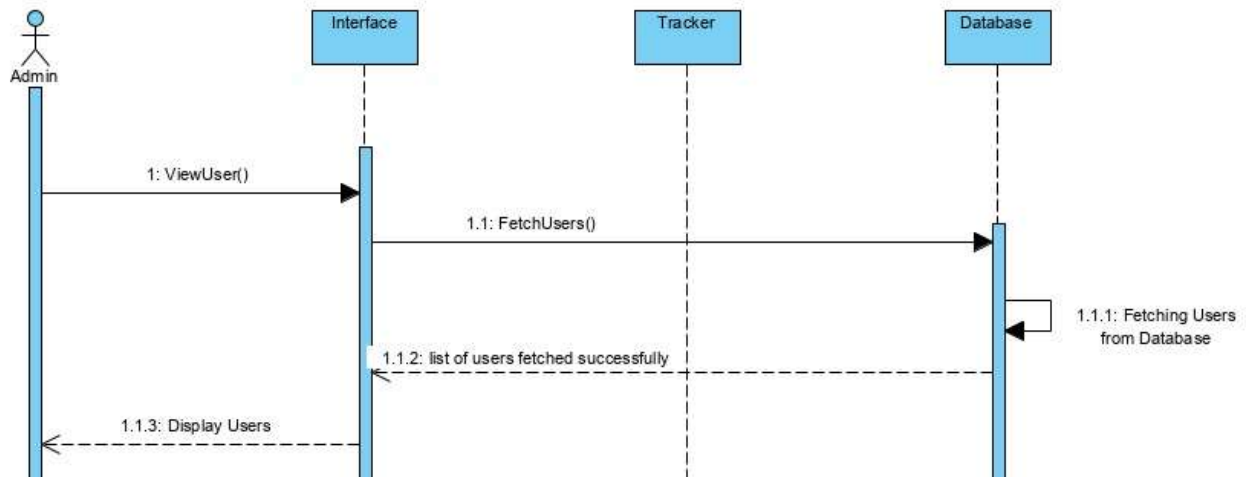


Figure 5.7 View Users

SSTRUM

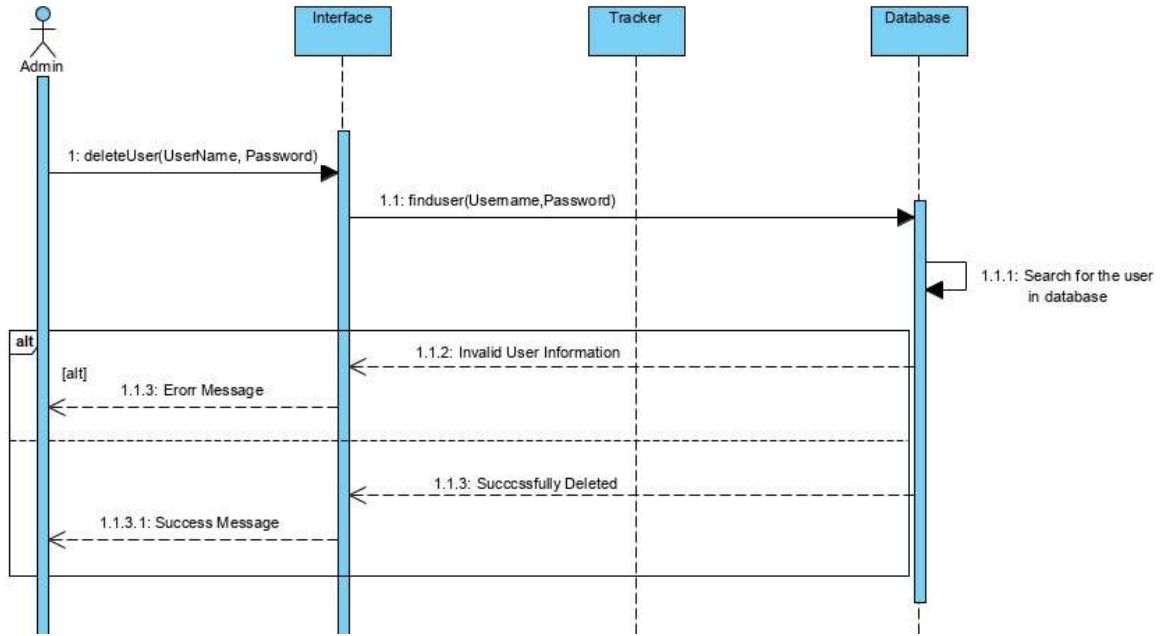


Figure 5.8 Delete User

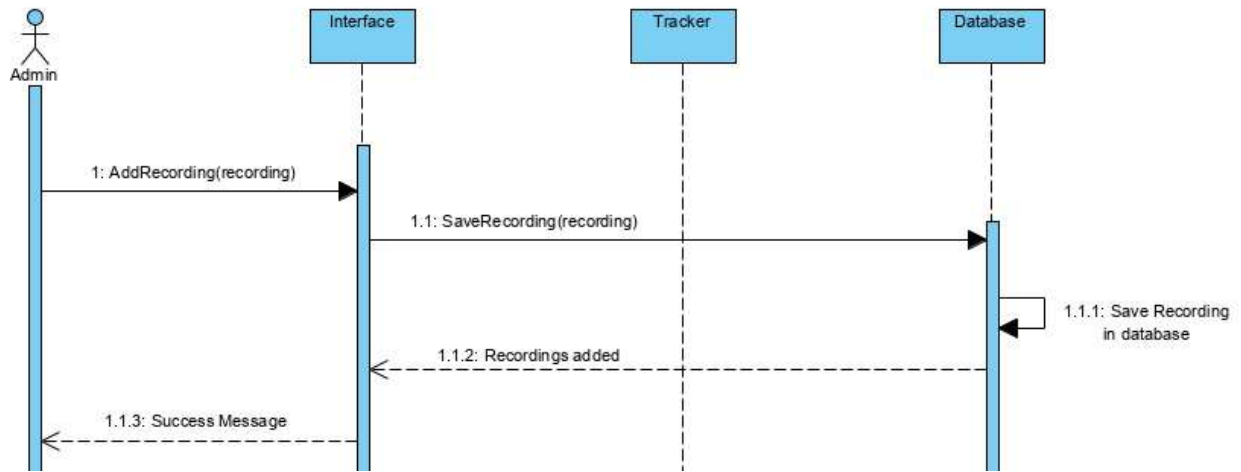


Figure 5.9 Add Recordings

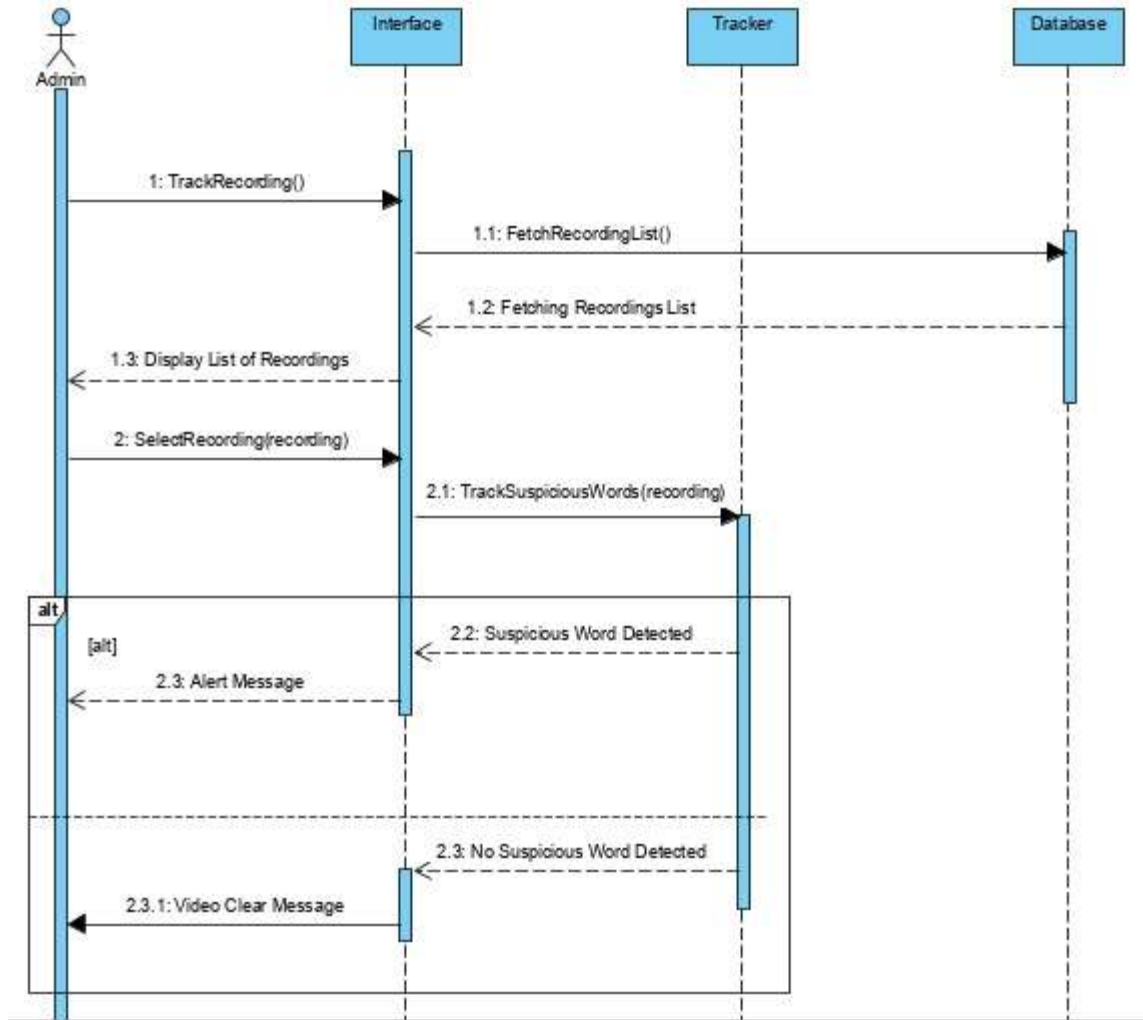


Figure 5.10 Track Recordings

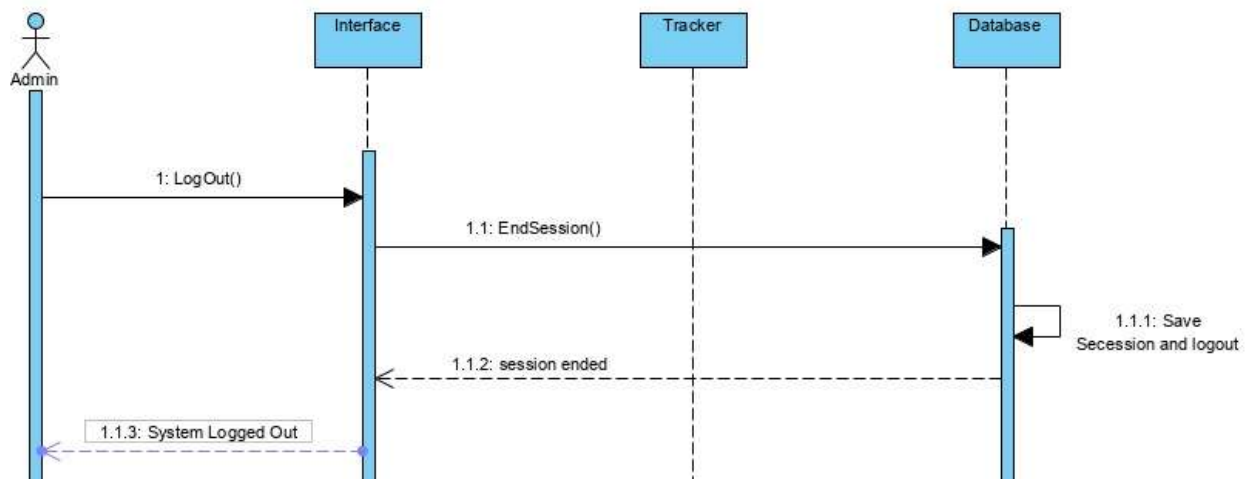


Figure 5.11 Logout

5.7 Data Design

5.7.1 Data Description

System saves user data in the SSTRUM 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 suspicious words. The audio uploaded by user will be matched with already provided dataset in the files. If any suspicious words will be found then the user will receive an alert message.

5.7.2 Data Dictionary

Objects	Attributes	Methods	Parameters
		AddUser()	Input: String, String
		RemoveUser ()	Input: String
		ViewUser()	Input: String
		SignIn()	Input: String, String
		SignOut()	
		AddRecording()	Input: Audio
		adminAuthentication()	Input: String Output: boolean
		userAuthentication()	Input: String Output: boolean
		SignIn()	Input: String,String
		SignOut()	
		AddRecording()	Input: Audio

		StartTracking()	Input: Audio
		TrackRecording()	Input: Audio
		MarkRecordingAsSuspicious()	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.

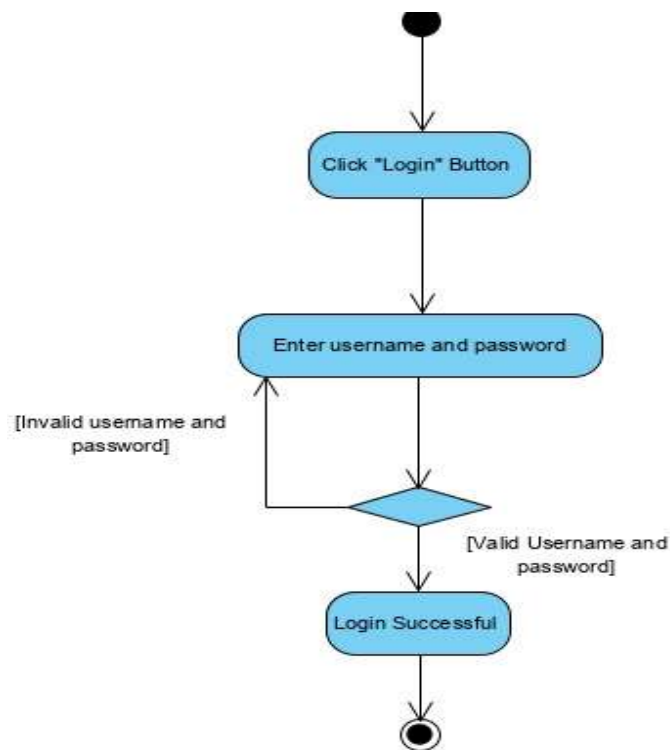


Figure 5.12 Login

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.

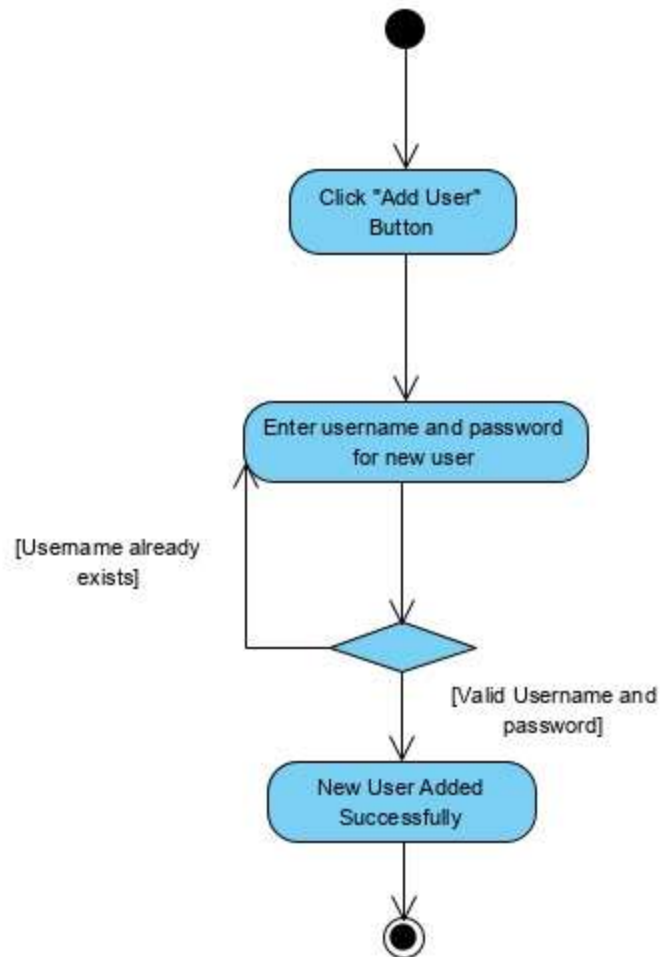


Figure 5.13 Add User

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.

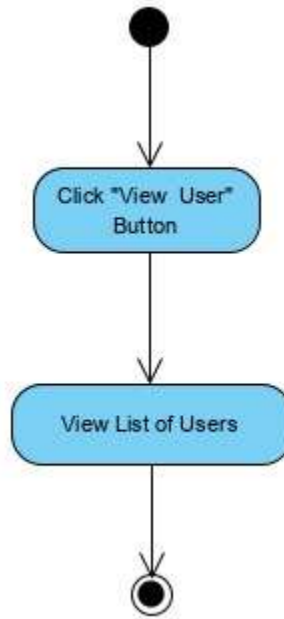


Figure 5.14 View Users

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.

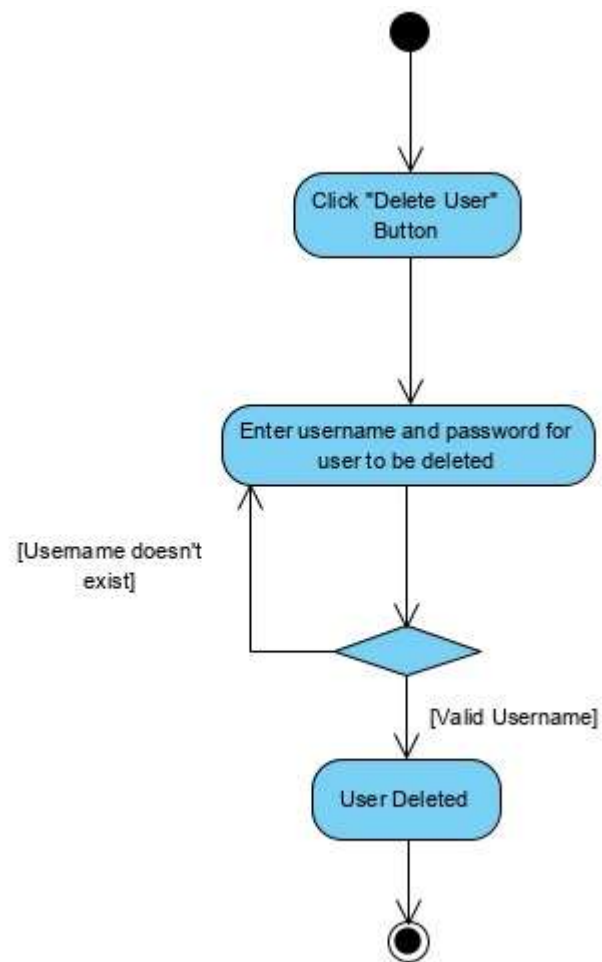


Figure 5.15 Delete User

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.

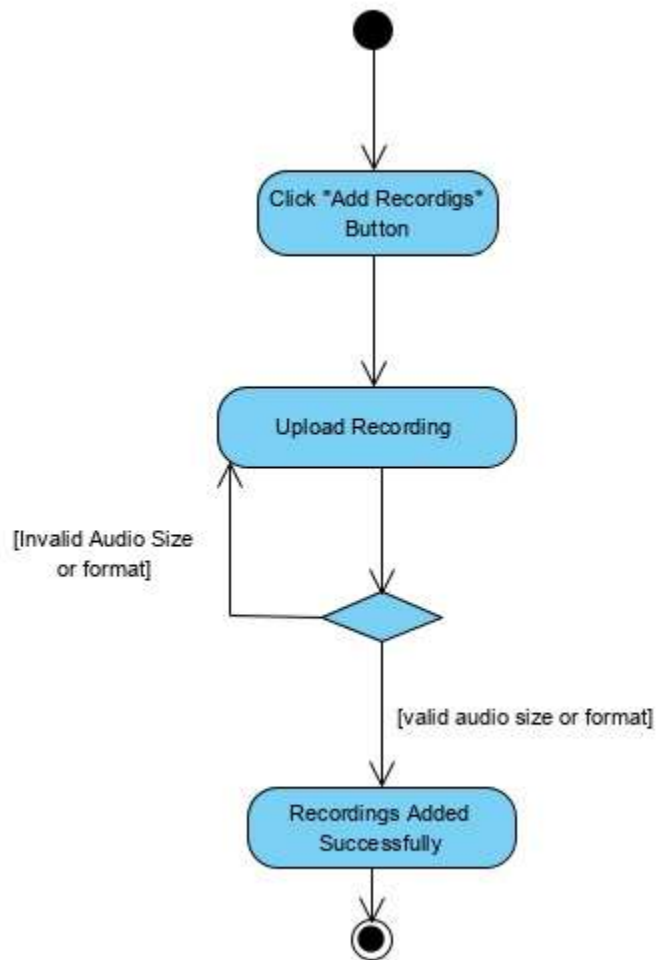


Figure 5.16 Add Recordings

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 suspicious word alert message will appear on the screen, else a prompt will appear declaring the recording to be safe.

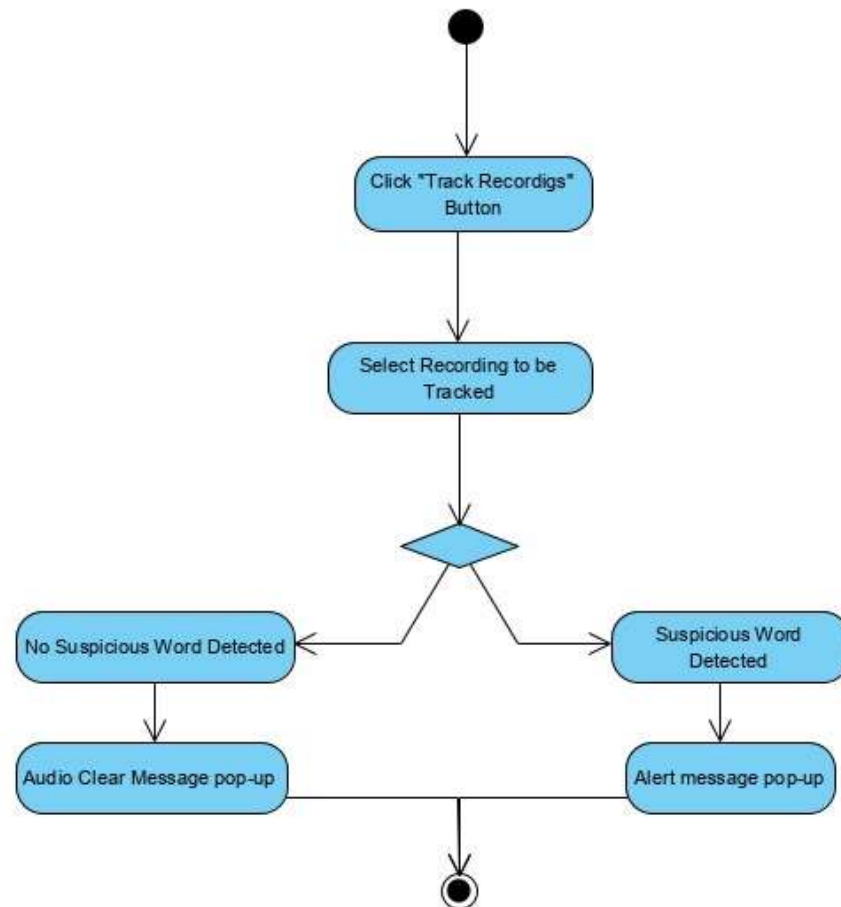


Figure 5.17 Track Recording

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

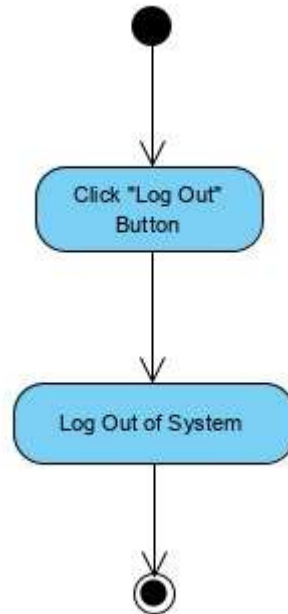


Figure 5.18 Log Out

5.9 Human Interface Design

5.9.1 Overview of User Interface

The users of the system will be able to login to the SSTRUM 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 SSTRUM 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 Images

1. Splash Screen:



Figure 5.19 Home screen

2. Main Screen

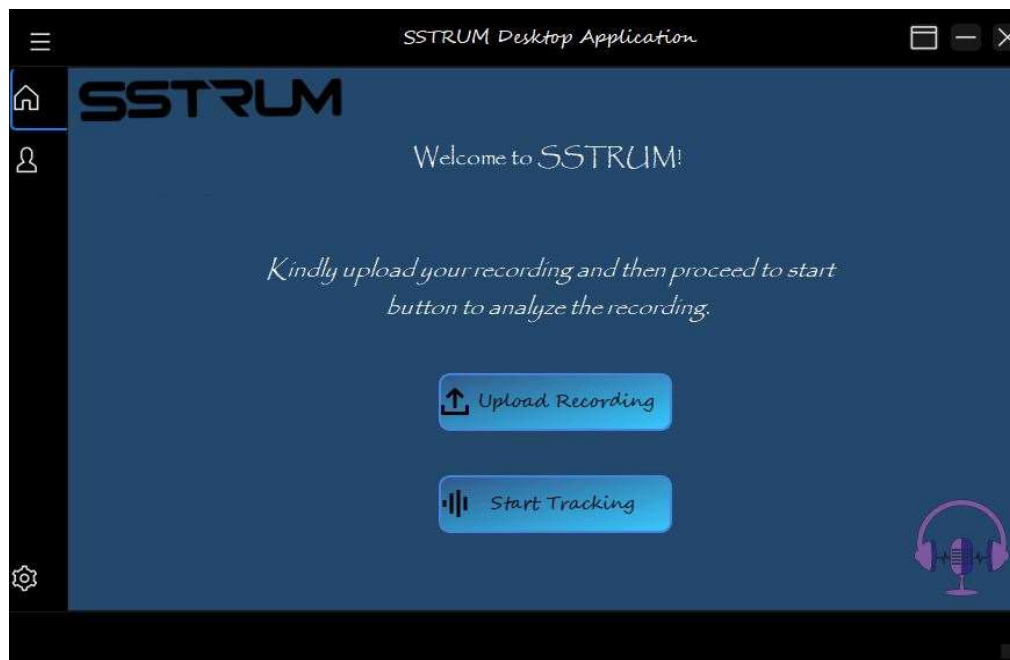


Figure 5.20 Main Screen

3. Admin Panel:



Figure 5.21 Admin Panel

4. Login Page:

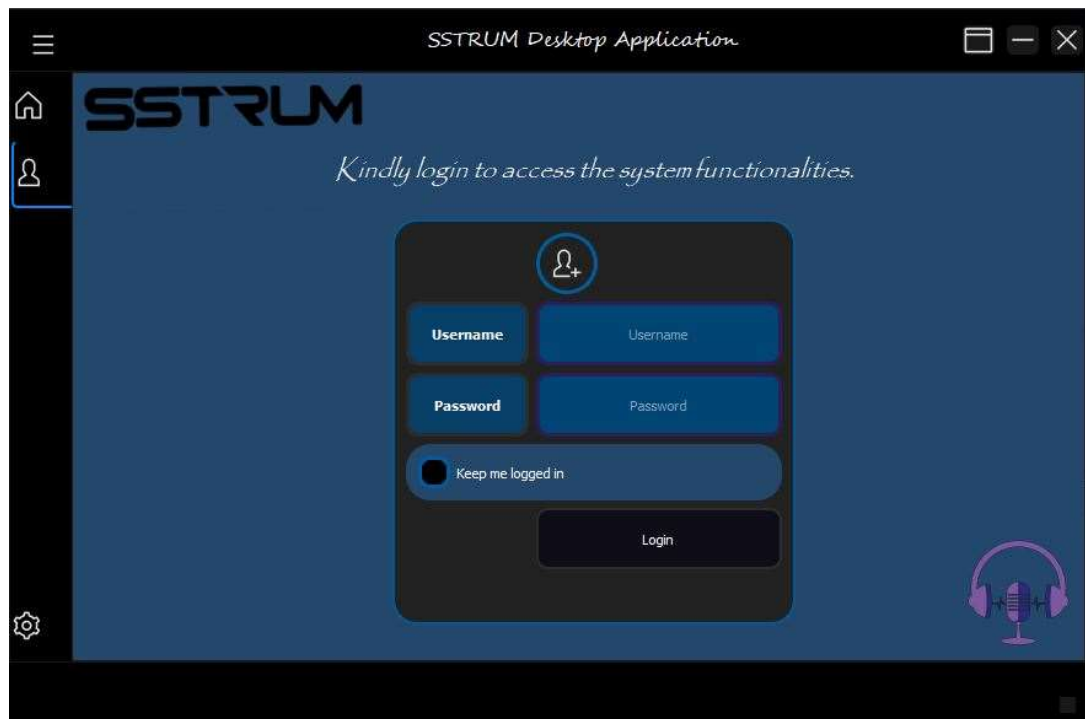


Figure 5.22 Login Page

5. Correct Username & Incorrect Password:

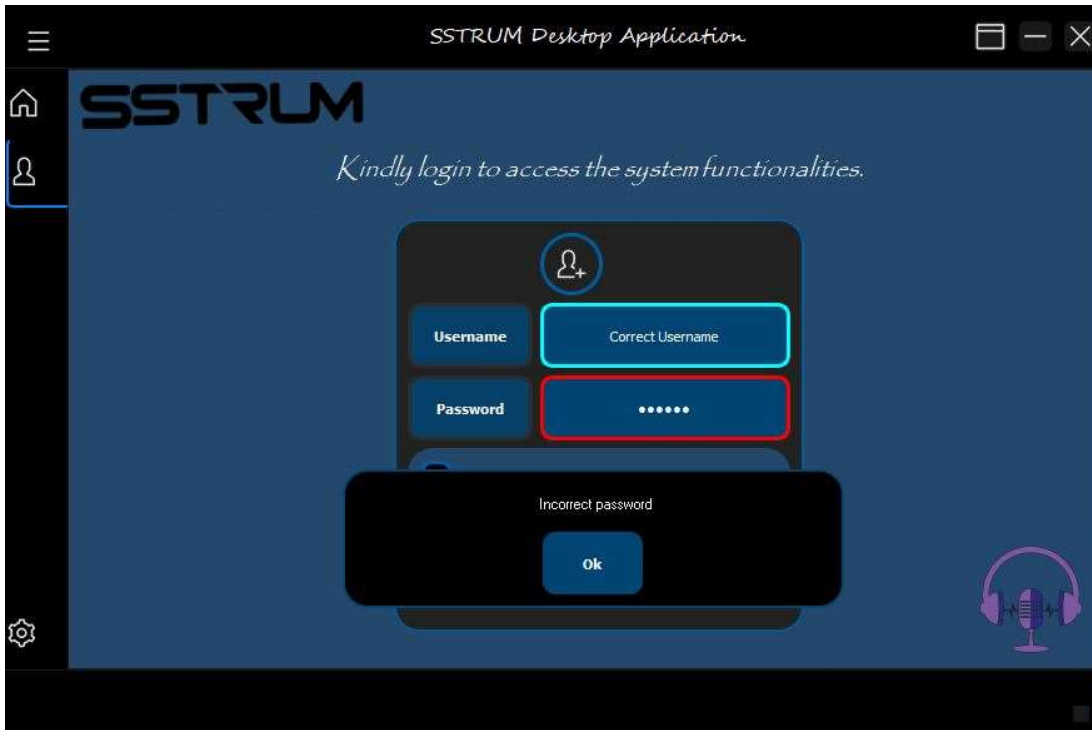


Figure 5.23 Incorrect Password

6. Correct Username & Correct Password:

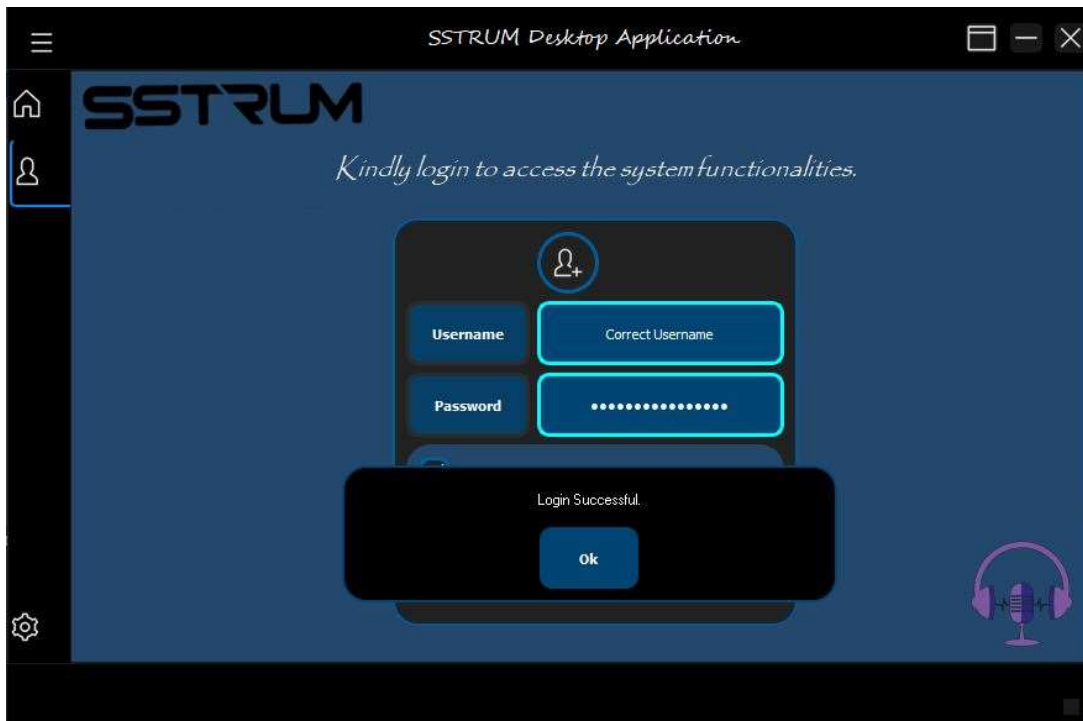


Figure 5.24 Correct Password

7. Empty Password:

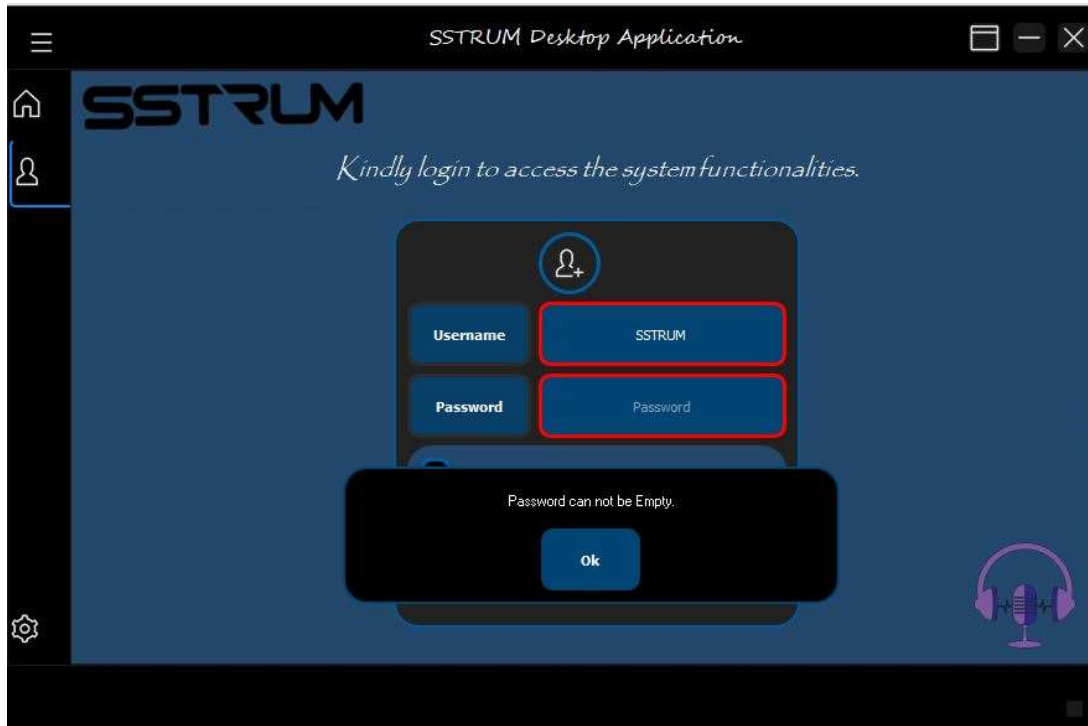


Figure 5.25 Empty Password

8. Incorrect Username & Incorrect Password:

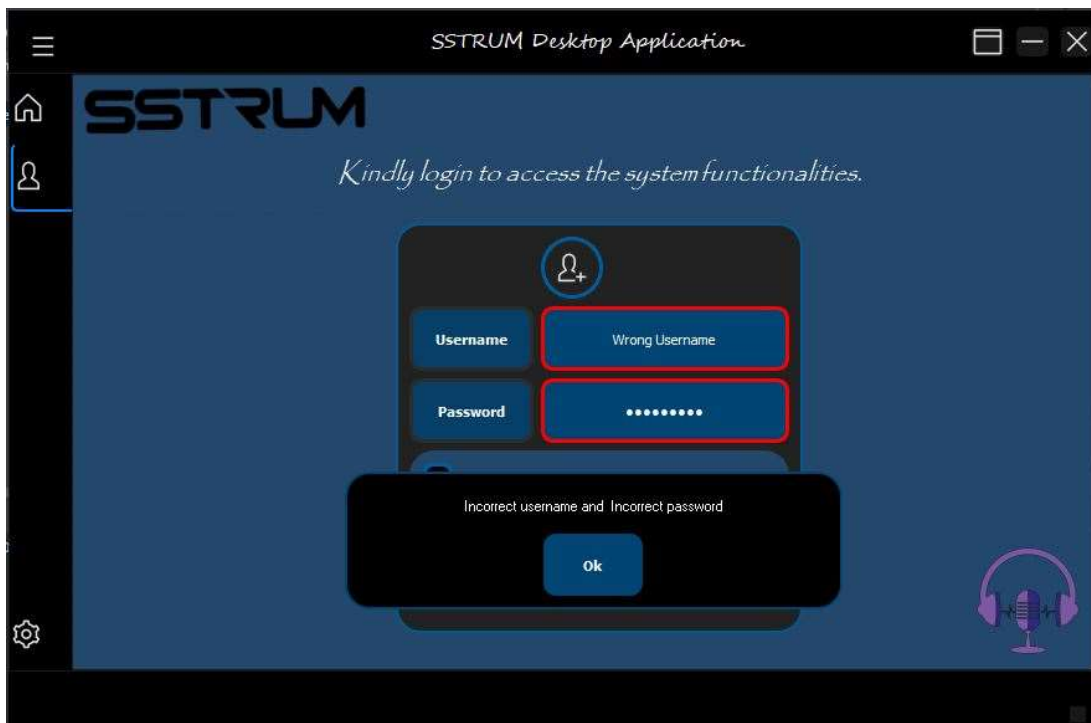


Figure 5.26 Incorrect Password and username

9. Empty Username & Empty Password:

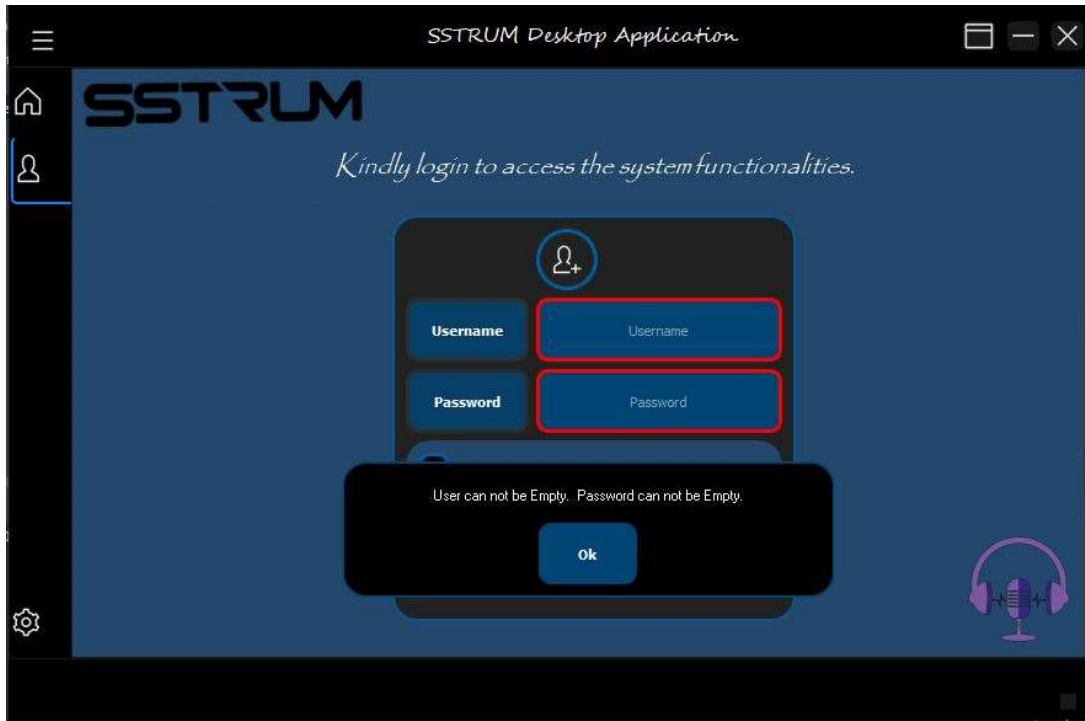


Figure 5.27 Empty Username & Empty Password

10. Empty Username:

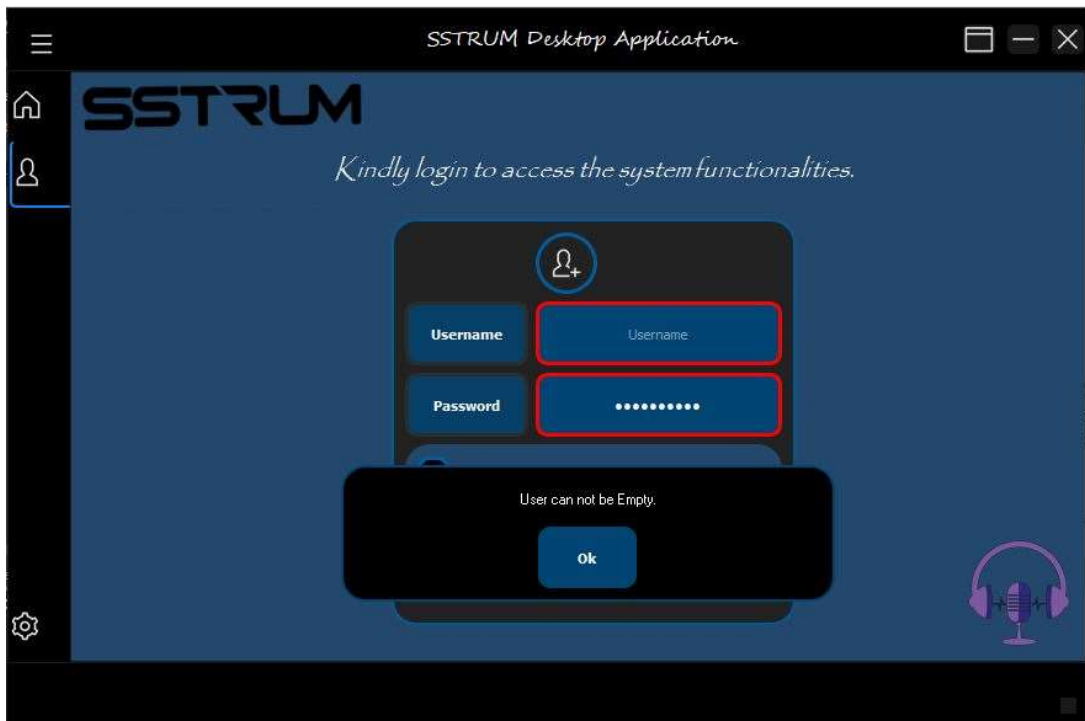


Figure 5.28 Empty Username

5.9.3 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, SSTRUM will start analysing the uploaded audios and match the words with dataset. If a suspicious 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

System Testing

6. Analysis and Evaluation

6.1. Introduction

This portion will cover the methodologies used to execute and manage "SSTRUM" testing process.

The test plan will ensure that SSTRUM is able to fulfill all the requirements.

6.2. Approach

All the test cases will be stated first and then analyzed on the basis of a report that will then be summarized to check for the SSTRUM 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 have to 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 has the ability to 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. SSTRUM should be able to detect suspicious calls.
8. The user should be able to receive proper alerts for suspicious 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 criterion to judge these tests.

1. Pre requisites 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 are 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 SSTRUM should be open
Input Values	Enter username and password then click “Login”
Steps	<ol style="list-style-type: none">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 SSTRUM.
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	<ol style="list-style-type: none"> 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 SSTRUM.
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	<ol style="list-style-type: none"> 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 SSTRUM.
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	<ol style="list-style-type: none"> 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	<ol style="list-style-type: none"> 1. Open the web portal. 2. Login as admin/user. 3. Click “Add Recording”. 4. Upload recording 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	<ol style="list-style-type: none"> 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 Suspicious Calls
Description	The system should be able to detect suspicious calls.

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	<ol style="list-style-type: none"> 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 Suspicious Calls

Test Case Number	08
Test Case Name	Alert Message
Description	The user should be able to receive proper alerts for suspicious 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	<ol style="list-style-type: none"> 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 suspicious.
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	<ol style="list-style-type: none"> 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

SSTRUM was aimed at simplifying the automatic speech recognition for suspicious speech tracking by making it user friendly. At the same time, in such systems, accuracy has to be taken in to account as it is of the primary importance. For this, thorough quality and accuracy check was performed regularly on our dataset so that we get the best possible accuracy.

Here we implemented our Pashto based recognition that contained 15 words, 10 non suspicious and 5 suspicious words. Training and testing parts of dataset were spread into two separate portions. Firstly, features were extracted using MFCC algorithm and then machine-learning algorithms were applied on the data.

For our system, we tried six different algorithms to check which algorithm supports our system the best. We have noted the accuracies for all the systems for the readers in order to make it easier for them to analyze how different models can perform under such situations.

Following is the table showing the accuracies of all the algorithms.

Algorithm	Accuracy
CNN	84.3%
SVM	70.5%
Naïve Bayesian	45%
RNN	64.8%
Multilayer perceptron	72.1%
Random Forest Classification	72.5%

Table 7.1 Accuracy of Algorithms

From the above results, it is clearly visible that the best accuracy could be attained by **Convolutional Neural Network**. Moreover, it was also noted that the time taken by CNN was also the shortest. Hence we used HMM along with CNN for our system.

By exploring different approaches for our system, we came to know that CNN can be very effective in automatically learning features directly from the dataset during the training process. Here we used MFCC for tuning of CNN, which definitely worked well. The results of our system demonstrate that well trained Convolutional Neural Network maximizes the performance in our system. Our proposed system was shown to improve and add value to previous approaches.

Further, for calculating the word error rate of our system, we tested all the 15 words in table 2 and table 3 for five different speakers. While having the same training and testing data, we tried different speakers in order to get the minimum WER. The following table shows the percentage error per word of every speaker.

Test Number	% error
1	0
2	0
3	13.33
4	20
5	13.33

Table 7.2 CNN error rate

From the above results we conclude that the total percentage error of our system is 15.7%. This could be further reduced by adding more data to our dataset, which is a goal to expand our prototype system.

Furthermore, a dedicated desktop application was also created with a very user friendly interface which was integrated with our system for easy tracking of audios. This application is working efficiently where users can add audios for analysis which is done by using the SSTRUM system at the backend. The accuracy of this application also came out to be similar to what we have presented in our thesis, making it an effective platform for suspicious speech tracking.

Chapter 8

Future Work

SSTRUM can be extended further to automate the process of speech tracking on large vocabulary. We will try to add more data marked as suspicious and non-suspicious to our project to cover as much Pashto language as possible. More the data, more accuracy could be achieved from the system. Moreover, to depend on it as an accurate device, we need to cover all the possible vocabulary of Pashto covering all the dialects. For the greater cause, this database will be open to being shared with researchers who want to work on any such system in the Pashto language. As of now, we have developed a desktop application, but we will try to move it to a standalone device, so that it could be deployed easily by security agencies.

6.1 Increasing Accuracy

Currently our system is based on 2535 utterances of 15 words by 169 different speakers. The accuracy of our system based on this system using CNN is 84.3%. However, further to improve this accuracy, more data is needed. We will add more data by increasing the number of speakers per word.

6.2 Enhancing Dataset

We have made a prototype of our idea based on 5 words marked as suspicious and 10 words marked as non-suspicious. However, to make it able to actually perform in a real time environment, maximum vocabulary of Pashto speech needs to be covered. We will focus on completely covering the whole vocabulary of Yousafzai dialect at first. After that, it could be further extended to other Pashto dialects.

6.3 Standalone Device

As of now, interface of SSTRUM is developed in PyQt5 making it a very user-friendly desktop application. Since we want to use this application mainly for security agencies, so a standalone device will be preferred to deploy it directly to their system. We will work on making a proper standalone device powered by Raspberry pi 4 and connected to an LCD screen.

Chapter 9

Conclusion

The project “Suspicious Speech Tracking Using Machine Learning” provides a prototype of the solution for automating the process of call tracking to detect any suspicious talk among terrorists. It will be a good way of efficiently doing the task and saving manpower making it very cost-effective. The unique feature of our work is the Pashto language. We have chosen the Pashto language since a lot of work has already been done in other local languages of Pakistan except for Pashto. According to our research, no dataset for Pashto words except for the counting was made. For the greater cause, this database will be open to being shared with researchers who want to work on any such system in the Pashto language.

Keeping in view the need for technology in every field, SSTRUM is a very smart solution of avoiding terror activities by completely automating the whole process. Our product has a very user-friendly interface with a response time of few seconds.

SSTRUM system can be used in many different security agencies including Police, Intelligence Bureau, Military Intelligence, and ISI by adding a dataset according to the need for the required job.

.

Chapter 10 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
- [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.
- [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.

[23] Marius, Popescu & Balas, Valentina & Perescu-Popescu, Liliana & Mastorakis, Nikos. (2009). Multilayer perceptron and neural networks. WSEAS Transactions on Circuits and Systems. 8.

[24] Cutler, Adele & Cutler, David & Stevens, John. (2011). Random Forests. 10.1007/978-1-4419-9326-7_5.

Appendix A

Plagiarism Report

thesis

ORIGINALITY REPORT

14%

SIMILARITY INDEX

7%

INTERNET SOURCES

6%

PUBLICATIONS

9%

STUDENT PAPERS

PRIMARY SOURCES

1Submitted to Higher Education Commission
Pakistan

Student Paper

2%**2**Tuka AlHanai, Wei-Ning Hsu, James Glass.
"Development of the MIT ASR system for the
2016 Arabic Multi-genre Broadcast
Challenge", 2016 IEEE Spoken Language
Technology Workshop (SLT), 2016

Publication

1%**3**

docplayer.net

Internet Source

1%**4**

coeal19.wixsite.com

Internet Source

1%**5**Submitted to Colorado Technical University
Online

Student Paper

<1%**6**Bakht Zada, Rahim Ullah. "Pashto isolated
digits recognition using deep convolutional
neural network", Heliyon, 2020

Publication

<1%