

A PROJECT REPORT
ON
“SMART CRIME REPORTER”

Submitted to
UNIVERSITY OF MUMBAI

In Partial Fulfillment of the Requirement for the Award of

BACHELOR’S DEGREE IN
COMPUTER ENGINEERING

BY

SHAIKH SHAKIR FAYYUMODDIN RIHANABI	16DCO78
SHAIKH MOHD AYUB MOHD ASGAR SAIRA BANO	15DCO69
MALDAR SAFA ALTAF AISHA	16DCO60

UNDER THE GUIDANCE OF
PROF. ABDUL SALAM SHAIKH



DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam's Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY

Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

2018-2019

AFFILIATED TO
UNIVERSITY OF MUMBAI

**A PROJECT II REPORT
ON**

“SMART CRIME REPORTER”

**Submitted to
UNIVERSITY OF MUMBAI**

In Partial Fulfillment of the Requirement for the Award of

**BACHELOR’S DEGREE IN
COMPUTER ENGINEERING**

BY

**SHAIKH SHAKIR FAYYUMODDIN RIHANABI
SHAIKH MOHD AYUB MOHD ASGAR SAIRA BANO
MALDAR SAFA ALTAF AISHA**

**16DCO78
15DCO69
16DCO60**

**UNDER THE GUIDANCE OF
PROF. ABDUL SALAM SHAIKH**



**DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam's Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206**

**2018-2019
AFFILIATED TO**



UNIVERSITY OF MUMBAI

Anjuman-i-Islam's Kalsekar Technical Campus

Department of Computer Engineering
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206



CERTIFICATE

This is certify that the project entitled
“ **SMART CRIME REPORTER** ”

submitted by

SHAIKH SHAKIR FAYYUMODDIN RIHANABI	16DCO78
SHAIKH MOHD AYUB MOHD ASGAR SAIRA BANO	15DCO69
MALDAR SAFA ALTAF AISHA	16DCO60

is a record of bonafide work carried out by them, in the partial fulfillment of the requirement for the award of Degree of Bachelor of Engineering (Computer Engineering) at *Anjuman-I-Islam's Kalsekar Technical Campus, Navi Mumbai* under the University of MUMBAI. This work is done during year 2018-2019, under our guidance.

Date: / /

PROF. ABDUL SALAM SHAIKH
Project Supervisor

PROF. KALPANA BODKE
Project Coordinator

PROF. TABREZ KHAN
HOD, Computer Department

DR. ABDUL RAZAK HONNUTAGI
Director

External Examiner

Acknowledgements

I would like to take the opportunity to express my sincere thanks to my guide **Abdus Salam Shaikh**, Assistant Professor, Department of Computer Engineering, AIKTC, School of Engineering, Panvel for his invaluable support and guidance throughout my project research work. Without his kind guidance & support this was not possible.

I am grateful to him/her for his timely feedback which helped me track and schedule the process effectively. His/her time, ideas and encouragement that he gave is help me to complete my project efficiently.

We would like to express deepest appreciation towards **DR. ABDUL RAZAK HONNUTAGI**, Director, AIKTC, Navi Mumbai, **Prof. TABREZ KHAN**, Head of Department of Computer Engineering and **Prof. KALPANA BODKE**, Project Coordinator whose invaluable guidance supported us in completing this project.

At last we must express our sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped me directly or indirectly during this course of work.

SHAIKH SHAKIR FAYYUMODDIN RIHANABI

SHAIKH MOHD AYUB MOHD ASGAR SAIRA BANO

MALDAR SAFA ALTAF AISHA

W

Project I Approval for Bachelor of Engineering

This project entitled *Smart Crime Reporter* by *Shaikh Ayub* is approved for the degree of *Bachelor of Engineering in Department of Computer Engineering*.

Examiners

1.

2.

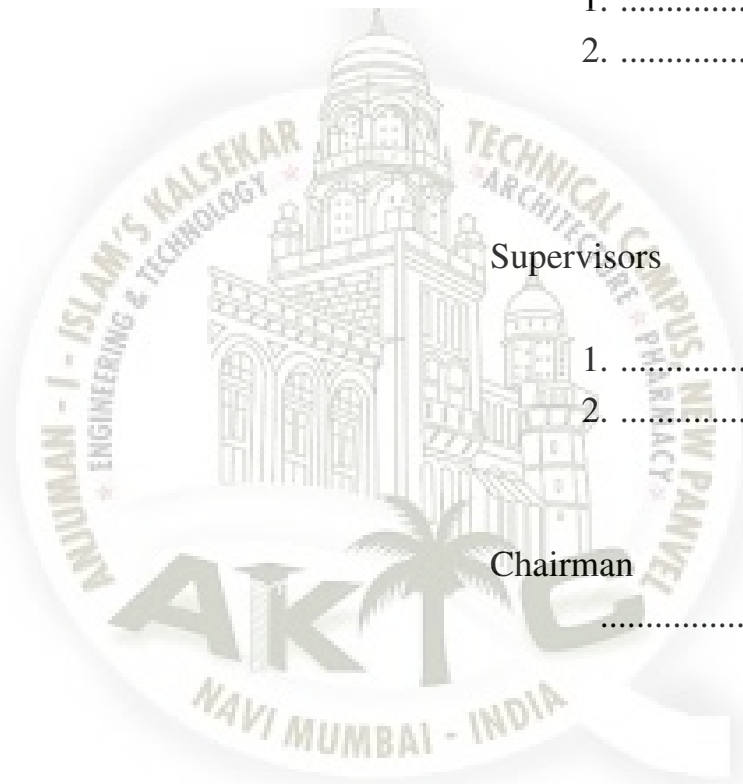
Supervisors

1.

2.


Chairman

.....



Declaration

I declare that this written submission represents my ideas in my own words and where others ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

The logo of AIKTC (Anjuman-e-Islam's Kalsekar Technical Campus) is a circular emblem. It features a central illustration of a large, ornate building with multiple domes and minarets, characteristic of Islamic architecture. The text around the circle includes 'ANJUMAN - I - ISLAM'S KALSEKAR' at the top, 'ENGINEERING & TECHNOLOGY' on the left, 'TECHNICAL CAMPUS' on the right, and 'NAVI MUMBAI - INDIA' at the bottom. The acronym 'AIKTC' is prominently displayed in the center of the emblem.

Shaikh Shakir Fayumoddin Rihanabi
16DCO78

Shaikh Mohd Ayub Mohd Asgar Saira Bano
15DCO69

Maldar Safa Altaf Aisha
16DCO60

ABSTRACT

The use of mobile Technology in Prevention of Crime has been increased across the world. However this is not the only case in India, because the People still have to go from one police station to another to lodge complaints against Crimes and to receive basic services from the Police.

The crimes that occurs are usually recorded manually in a particular Register called Police Occurrence Book(OB) which creates problems while determining the accurate crimes and the type of Crimes occurred till date which later proved to be major hurdle in Crime Prevention for the Indian Police.

It is often seen that people of our society does not report about the crime until and unless if it is happened to them or to their family members jut because they don't want to visit the police station and waste their precious time to report about the same, so the main motivation behind writing this paper is for the betterment of our society, many people hesitate to report about the crime but this study involves about developing an android App which will help people to report about crime freely without visiting the police station.

Keywords: Crime, Voice Recognition, Keywords, GPS Tracking, Reporting.

Shaikh Shakir Fayumoddin Rihanabi
16DCO78

Shaikh Mohd Ayub Mohd Asgar Saira Bano
15DCO69

Maldar Safa Altaf Aisha
16DCO60

B.E(Computer Engineering)
University of Mumbai

Contents

Acknowledgement	iii
Project I Approval for Bachelor of Engineering	iv
Declaration	v
Abstract	vi
Table of Contents	ix
1 Introduction	2
1.1 Purpose	2
1.2 Project Scope	2
1.3 Project Goals and Objectives	3
1.3.1 Goals	3
1.3.2 Objectives	3
1.4 Organization of Report	3
2 Literature Survey	4
2.1 Keyword Spotting for Google assistant Using Contextual Speech Recognition	4
2.1.1 Advantages of Paper	4
2.1.2 Disadvantages of Paper	4
2.1.3 How to overcome the problems mentioned in Paper	4
2.2 The Adoption of Mobile Technology as a tool for situational Crime Prevention	5
2.2.1 Advantages of Paper	5
2.2.2 Disadvantages of Paper	5
2.2.3 How to overcome the problems mentioned in Paper	5
2.3 Implementaion of GPS for location tracking	6
2.3.1 Advantages of Paper	6
2.3.2 Disadvantages of Paper	6
2.3.3 How to overcome the problems mentioned in Paper	6
2.4 Technical Review	7
2.4.1 Advantages of Technology	7
2.4.2 Reasons to use this Technology	7

3	Project Planning	8
3.1	Members and Capabilities	8
3.2	Roles and Responsibilities	8
3.3	Assumptions and Constraints	8
3.3.1	Assumptions	8
3.3.2	Constraints	9
3.4	Project Management Approach	9
3.4.1	Requirement Analysis	9
3.4.2	System Design	10
3.4.3	Implementation	10
3.4.4	Integration and Testing	10
3.4.5	Deployment of system	10
3.4.6	Maintenance	10
3.5	Ground Rules for the Project	11
3.6	Project Budget	11
3.7	Project Timeline	11
4	Software Requirements Specification	12
4.1	Overall Description	12
4.1.1	Product Perspective	12
4.1.2	Product Features	12
4.1.3	User Classes and Characteristics	12
4.1.4	Operating Environment	13
4.1.5	Design and Implementation Constraints	13
4.2	System Features	13
4.2.1	System Feature	13
4.3	External Interface Requirements	14
4.3.1	User Interfaces	14
4.3.2	Hardware Interfaces	14
4.3.3	Software Interfaces	14
4.3.4	Communications Interfaces	15
4.4	Nonfunctional Requirements	15
4.4.1	Performance Requirements	15
4.4.2	Safety Requirements	15
4.4.3	Security Requirements	15
5	System Design	16
5.1	System Requirements Definition	16
5.1.1	Functional Requirements	16
5.1.2	System requirements (non-functional requirements)	19
5.1.3	Safety Requirements	20
5.1.4	Security Requirements	20

5.2	System Architecture Design	20
5.3	Sub-system Development	21
5.3.1	Speech Recognition	21
5.3.2	Keywords Matcher	22
5.3.3	Image Processing	22
5.3.4	Location Detector	22
5.3.5	Report	22
5.4	Systems Integration	23
5.4.1	Class Diagram	23
5.4.2	Sequence Diagram	24
5.4.3	Component Diagram	25
5.4.4	Deployment Diagram	26
6	Implementation	27
6.1	Select User	27
6.2	Registration for User/Police	29
6.3	Login for User/Police	32
6.4	User Panel	34
6.5	Safety Tips	36
6.6	Report Activity	38
6.7	Weapon Detector	42
6.8	Police Panel	44
6.9	Report List	46
6.10	Crime Location	48
7	System Testing	50
7.1	Test Cases and Test Results	50
7.2	Sample of a Test Case	50
7.2.1	Software Quality Attributes	55
8	Screenshots of Project	56
8.1	Smart Crime Reporter	57
9	Conclusion and Future Scope	62
9.1	Conclusion	62
9.2	Future Scope	62
	References	63
	Achievements	63

List of Figures

3.1	Waterfall Model	9
5.1	Use case of Smart Crime Reporter	17
5.2	Data flow Diagram Level 0	18
5.3	Data flow Diagram Level 1	18
5.4	Data flow Diagram Level 2a	19
5.5	Data flow Diagram Level 2b	19
5.6	Entity-Relationship Diagram	20
5.7	System Architecture of Smart Crime Reporter	21
5.8	Modular Diagram of Speech Recognition	22
5.9	Modular Diagram of Keywords matcher	22
5.10	Modular Diagram of Image Processing	22
5.11	Modular Diagram of Location Detector	22
5.12	Modular Diagram of Report	23
5.13	Class Diagram	23
5.14	Sequence Diagram of User	24
5.15	Sequence Diagram of Police	24
5.16	Component Diagram	25
5.17	Deployment Diagram	26
6.1	Select User	27
6.2	User Registration	29
6.3	User Login	32
6.4	{User Panel	34
6.5	Safety Tips	36
6.6	Report Activity	38
6.7	Weapon Detector	42
6.8	Police Panel	44
6.9	Report List	46
6.10	Crime Location	48
7.1	Login Page	51
7.2	Object Detection	53
7.3	Fetching Location	54

9.1	Paper Presentation Certificate- Ayub Shaikh	65
9.2	Paper Presentation Certificate- Shaikh Shakir	66
9.3	Paper Presentation Certificate- Safa Maldar	67



List of Tables

3.1	Table of Capabilities	8
3.2	Table of Responsibilities	8
7.1	Test Cases Table	50



Chapter 1

Introduction

The Study aimed at developing a Mobile Application that will help the people to report crimes to the nearest police station without any fear of Victimization and through which the police can reach to the Crime spot as earliest as possible. In many situation it happens people cannot use their phone to inform about the crime happening around them however this study involved in developing an App which will recognize the Keywords such as Help from the User and the app will sense the keyword through the use of Machine Learning and an Alert will be initiated in the App and it will be sent to the nearest police station with the location in which the crime has been Occurred

1.1 Purpose

The main objective of our system is to develop an application which will be beneficial for our society to report about crime easily just like on their finger tips, the main agenda is to make the application user-friendly so that even a common user without much knowledge about technologies can use it easily and efficiently, with help of this application even our police department will be benefited and can reach the crime spots as early as possible.

1.2 Project Scope

Our project is based on consideration that emergency may occur at any point of time thus using voice recognition or capturing image of crime spot is the most efficient way to seek a help from areas local police stations. Our project promises to replace the manual work of reporting to police station with an automated and quicker approach using natural language processing and machine learning and we are trying to implement using existing algorithms and APIs.[2]

1.3 Project Goals and Objectives

1.3.1 Goals

Our Ultimate goal is aimed at developing a Mobile Application that will help the people to report crimes to the nearest police station without any fear of Victimization and through which the police can reach to the crime spot as earliest as possible.

1.3.2 Objectives

The main objective of our system is: • To provide flexibility for the public to use to while in emergency. • To generate an automated call with location details from an input as spoken keyword or captured image. • To store details of list of police stations and update their contact information regularly.

1.4 Organization of Report

Chapter 2: We have discussed about various paper that we have referred for our project. We have mentioned the description , pros and cons and how the overcome the problem under every paper. A total of three paper have been referred.

In Chapter 3: We have discussed about the requirement analysis under it we have consider about the requirement the platform requirement supporting the OS of the software and hardware requirement along with the feasible study.

In Chapter 4: This chapter basically deals with the diagrammatic representation of the modules. We can see system design and various other modules

In Chapter 5: Here we have seen the methodology here we have explain the project in detail by dividing into module. Various module of discount card and hierarchy business are explained with the help of few diagram.

In Chapter 6: we have discussed about the implementation details the assumption and dependencies this part contains details of the implementation of methodology that we discuss earlier.

In Chapter 7: we have shown the test cases and result along with analytic discussion this part contained the result of the output of the project.

In Chapter 8: we have concluded the whole project and future scope along with the limitation followed up by reference and chapter 9 with Appendix.

Chapter 2

Literature Survey

2.1 Keyword Spotting for Google assistant Using Contextual Speech Recognition

We are proposing a system which uses speech to recognize contextual automatic speech recognition (ASR) for Voice activated devices. The idea is to spot the different keywords that is already stored in our database which later can be triggered as soon as it recognizes those key-words. Here we are using Keyword Spotting System (KWS) that runs on all voice activated devices, it is common that a KWS system is run onto the device in order to quickly detect a trigger phrase (e.g. “help Robbery”). After the trigger phrase is detected, the audio corresponding to the voice command that follows is streamed to the server.

2.1.1 Advantages of Paper

- a. Here the keywords are easily spotted using keywords spotting system.
- b. The speech is recorded automatically as soon as the keywords are spoken using Automatic Speech Recognition.

2.1.2 Disadvantages of Paper

- a. Many a times it happens Keywords are not Pronounced Properly also it is not interpreted as the actual word
- b. Due to Noisy Environment system is not able to listen the spoken words.

2.1.3 How to overcome the problems mentioned in Paper

- a. We are Using NLP (Natural Language Processing) if user fails to speak the correct keywords NLP helps to adapt the words which we used in our data sets and compare it with the words spoken at crime scene .

2.2 The Adoption of Mobile Technology as a tool for situational Crime Prevention

The use of mobile Technology in Prevention of Crime has been increased across the world. However this is not the only case in India, because the People still have to go from one police station to another to lodge complaints against Crimes and to receive basic services from the Police. The crimes that occurs are usually recorded manually in a particular Register called Police Occurrence Book(OB) which creates problems while determining the accurate crimes and the type of Crimes occurred tiil date which later proved to be major hurdle in Crime Prevention for the indian Police.

2.2.1 Advantages of Paper

- a. The use of Mobile technology to report about crimes comes into handy as user does not have to visit the Police station.
- b. The mobile technology will help the Police to reach at crime spot early as they do not get the update immediately about the crime.

2.2.2 Disadvantages of Paper

- a. People often spread fake news without confirming it with any authentic firm whether its True or not.
- b. The user often feels uncomfortable with news that spread around social apps without verifying news which brings chaos.

2.2.3 How to overcome the problems mentioned in Paper

- a. The System must be designed in such a way that it should Track every User who Lodges complaint because often people Bully the Police.

2.3 Implementaion of GPS for location tracking

Global Positioning Satellites network is known to have offered users with many applications especially in the area of tracking. Currently, fisherman who lost their way in the sea could locate their position by using GPS receiver installed on their boat. They could acquire a coordinate and call for a rescue crew to save them with reference to the GPS coordinate. GPS is used for navigation and provides continuous and timing information position of things anywhere in the world under any weather condition GPS consist nominally of 24 operational satellites orbiting the earth at very high altitude Satellites send signals to the GPS receiver to locate the exact position. These satellites are constantly monitored to make sure that they are working properly. The GPS parts consist of three segments which are user, space and control.

2.3.1 Advantages of Paper

- a. Stand alone global positioning system receivers are widely used nowadays to accurately locating one's position.
- b. By using stand alone GPS receivers the distance between two locations on earth can also be measured

2.3.2 Disadvantages of Paper

item It requires an Active Internet connection to connect to GPS so as to fetch location. A handset must have a Smart phone with GPS feature so as to fetch exact location.

2.3.3 How to overcome the problems mentioned in Paper

- a. The feature of GPs must be present so as to use the Technology to fetch exact latitudes and Longitudes.

2.4 Technical Review

The technologies that we are using in our project are as follows:-

1. Android, 2. Firebase, 3. Google Places.

2.4.1 Advantages of Technology

- a. Android is Renowned for its power, speed, Scalability and performance, Android is a highly utilitarian feature-packed platform that gives you the opportunity to develop tremendous apps with great ease. Let us go through a few of the major advantages of selecting Android platform for your app development
- b. The Firebase is a Backend-as-a-Service (BaaS) that offers the developers a wide spectrum of tools and services to develop high-quality apps at a much faster pace.
- c. Google Maps provides the layout of roads, the locations of cities and towns, state boundaries, geographical features, restaurant reviews and satellite images. Google provides the Street View perspective, allowing you to see houses, store-fronts and points of interest from a driver's point of view. In addition, Google Maps has indoor maps of some airports, museums and other facilities

2.4.2 Reasons to use this Technology

- a. Android provides a rich application framework that allows us to build innovative apps and games for mobile devices in a Java language environment
- b. The image detection feature will capture the footage at the crime scene so that police will come to know about the culprit with their Weapons so they can come accordingly.

Chapter 3

Project Planning

3.1 Members and Capabilities

Table 3.1: Table of Capabilities

SR. No	Name of Member	Capabilities
1	Shaikh Shakir	Android Programming
2	Shaikh Ayub	Documentation
3	Safa Maldar	Presentation

Work Breakdown Structure

3.2 Roles and Responsibilities

Table 3.2: Table of Responsibilities

SR. No	Name of Member	Role	Responsibilities
1	Shaikh Shakir	Team Leader	Android Programming
2	Shaikh Ayub	Team Member	Documentation
3	Safa Maldar	Team Member	UI Design

3.3 Assumptions and Constraints

3.3.1 Assumptions

It is essential to maintain a crime free environment and keep track on criminal activities held in different location for personal as well as for mass benets we found it quite difficult to report those activities by manual process. Our proposed system is aimed at developing a mobile application that will help the people to report crimes to the nearest police station without any fear of Victimization and through which the police can reach to the crime spot as earliest as possible.

3.3.2 Constraints

We make schedule for our project to complete it on time, based on different aspects that are required in our project. The cost of the project that is required for completing the project. Different quality attributes in projects and resources required in project. We tried minimizing risk and tolerance in our project.

3.4 Project Management Approach

In our project we had used Waterfall Model for implementing all the phases successfully. In this model, the software development activity is divided into different phases and each phase consists of series of tasks and has different objectives. Here one phase starts only when the previous phase is complete. Because of this nature, each phase of waterfall model is quite precise well defined. Since the phases fall from higher level to lower level so This model is best suitable for our project. It maintains a systematic step wise approach.

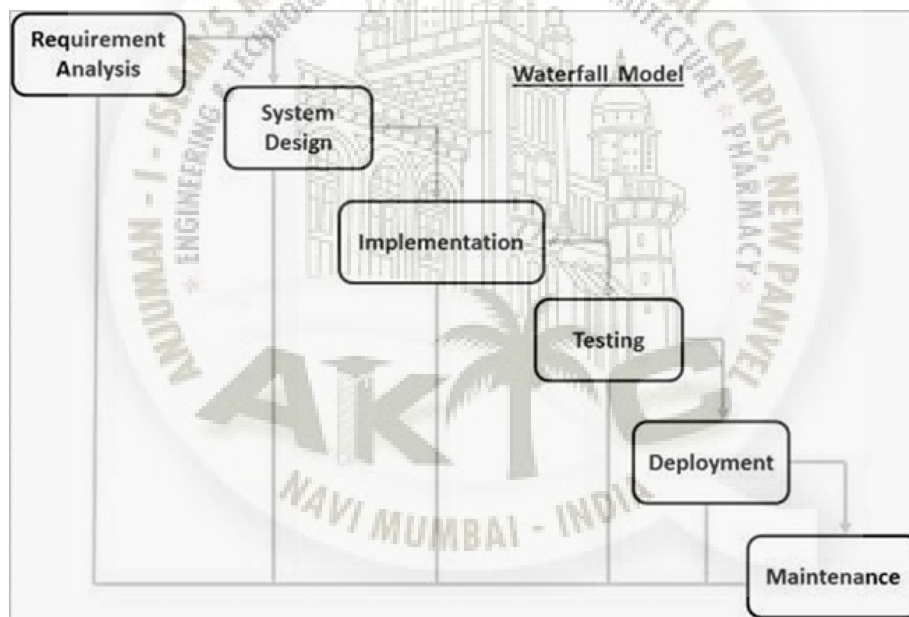


Figure 3.1: Waterfall Model

3.4.1 Requirement Analysis

The initial phase of any project is the requirement gathering. We need to gather proper information about our project. Analyze and understand the requirement and documented in a requirement specification document. We have visited various Crime reporting applications and analyzed many Hierarchy which will help gradually to develop our project with respect to Business as well.

3.4.2 System Design

The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture. We have design our App in such a way that it is very easy to use by any user and navigate the App without any hassle.

3.4.3 Implementation

With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing. The aim of the unit testing phase is to check whether each module is working properly or not.

3.4.4 Integration and Testing

All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures. The aim of the unit testing phase is to check whether each module is working properly or not.

3.4.5 Deployment of system

Before Deploying our Application first will check each functionality Once the functional and non-functional or not after that testing is done functional and Non functional testing is done, the product is deployed, in the customer environment or released into the market.

3.4.6 Maintenance

There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

3.5 Ground Rules for the Project

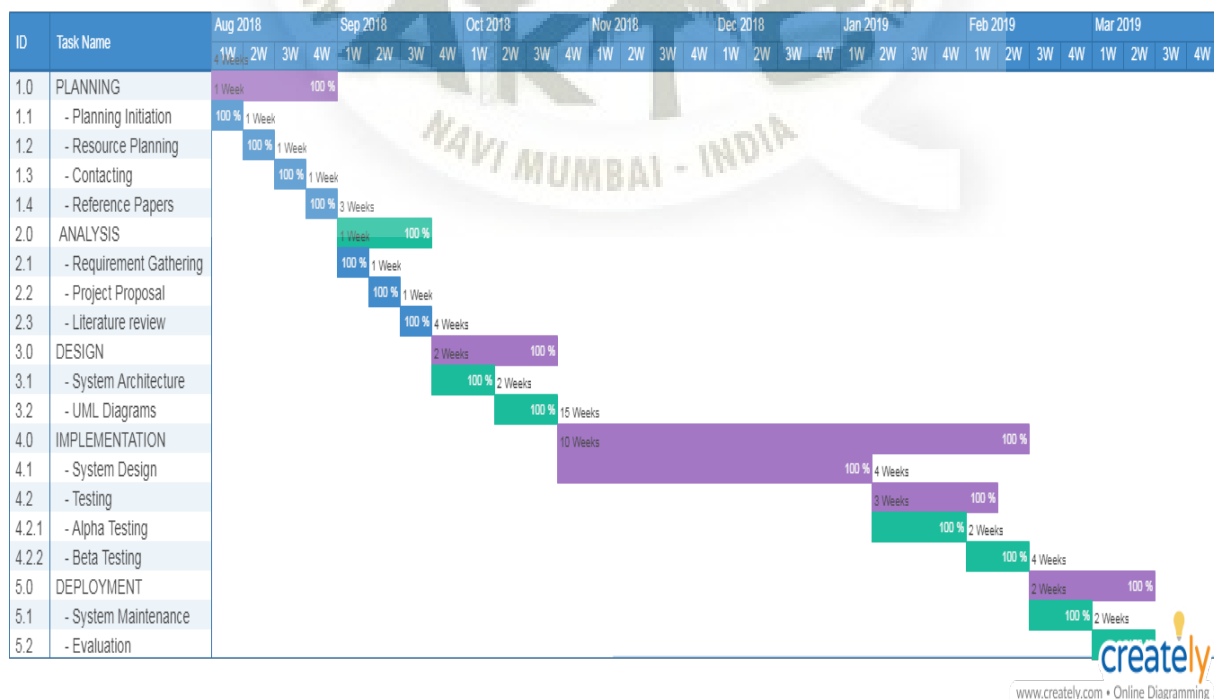
1. We treat each other with respect.
2. We intend to develop personal relationships to enhance trust and open communication.
3. We value constructive feedback. We will avoid being defensive and give feedback in a constructive manner.
4. As team members, we will pitch in to help where necessary to help solve problems and catch-up on behind schedule work.
5. Additional meetings can be scheduled to discuss critical issues or tabled items upon discussion and agreement with the team leader.
6. One person talks at a time; there are no side discussions.

3.6 Project Budget

The budget for the Project is very low as we are using Open Source tools and Services, following are the technologies that we have used :

1. Operating System: Linux Mint (Open Source), Windows, Mac
2. IDE: Android Studio: (Open Source)
3. Firebase Cloud Services.

3.7 Project Timeline



Chapter 4

Software Requirements Specification

4.1 Overall Description

4.1.1 Product Perspective

The perspective of the project is to create an Application which will help people in the society to report crimes easily, we are also focusing on making a better UI of our App which is quite convenient and will help in the better functioning and easy to navigate. It will be useful to each and everyone in the society who is ready to help someone by seeing them in trouble, this App will also help our Police department to reach at the crime spot as soon as possible with the help of location feature which is built in our App.

4.1.2 Product Features

The major feature of this app is that it will instantly report the crime by taking pictures on the Crime scene by spotting different weapons if present at the crime area. We have used an Algorithm which track the nearest Police station where the crime is actually committed and also with the Help of Google Places it will send the exact location to the police department about the crime. It will also send a report to the Police department about the crime committed, if the nearest police station fails to report.

4.1.3 User Classes and Characteristics

This Application can be used by Everyone except children, The user should have minimum knowledge that he/she can read the guidelines given on the app which will help them to navigate it properly. This App is mainly for those people who do not wants to fall in trouble or who are scared to report about crimes just because they do not have time for it, often people never report about crimes even if they are eye witness for it the main reason is they do not want them to visit the Police station for such people this App will prove to be very handy and it will help our department in a very good way.

4.1.4 Operating Environment

The environment in which the software will operate is Android and the hardware platform on which the software will run will be any Android based Smart Phone with GPS enabled. The Android Version should be Greater than 5.1

4.1.5 Design and Implementation Constraints

The major challenge that will hurdle the development of the system is the Voice Recognition, the app should respond to the Data sets that we have set, we have kept different keywords for each type of crime so it should trigger the App as soon as it hears the keywords. Another constraint would be the internet connectivity, if there is no internet connection App will not run. Also capturing images of the crime spot is also a difficult task.

4.2 System Features

The main feature of our system is to report about crimes fast and accurately, many a times it happens people fail to explain the crime spot to the police station but our App uses Google Places which will accurately find the co-ordinates and location will be sent. Many a times it happens at a crime scene where User cannot talk over the phone and explain about the situation where he is in, feature of Voice recognition will help at such a panic situation which triggers the App automatically as soon as it hears the keywords of the Crimes.

4.2.1 System Feature

1. Voice Recognition
2. GPS Location
3. Weapons Detection

Description and Priority

1. Voice Recognition

This feature helps to recognize the different keywords which is maintained in Database as soon as it hears it Trigger the App.

2. GPS Location

GPS Location is responsible for determining the Exact location of the crime scene which will be sent to Police side and hence will prove use full for the department to reach on time.

3. Weapons Detection

Weapons Detection is a feature which will detect the different objects at crime scene and detect it accurately if any weapon is found.

Stimulus/Response Sequences

1. User will login into the system. User can either be a Normal User or a Police.
2. After login User Can either report about Crimes or can read Safety Tips
3. If there is a Crime committed it will send the location to Police.
4. Application will Automatically capture the images in the crime scene it will detect weapons if there is any at Crime scene.
5. At the end Police can read the report.

Functional Requirements

1. The user should be able to login into the system.
2. The User can Speak keywords of the Crime to report, the moment keyword is spoken App will be triggered.
3. The GPS location will help to locate the crime spot and same will be shared with the police Side.
4. Weapon detector will help to detect the objects in the crime scene and it will be capture through the Phone's Camera.

4.3 External Interface Requirements

4.3.1 User Interfaces

1. User will be able to register and Login to the system.
2. After the Login is done session is maintained.
3. The Police will receive an alarm if Crime is reported by any User who uses the Application
4. The
5. The user will have the Analysis of the Hierarchy maintained.

4.3.2 Hardware Interfaces

Android enabled device: The android enabled device should have android version above 5.0. In order for the smooth functioning of the application the android device must have at least 512 MB of ram and at least 500 MB of free storage on device. The application can also function on a tablet device.

4.3.3 Software Interfaces

1. Operating System: Android above 4.4.
2. Database: firebase cloud Services .
3. Tools: Android studio IDE.

4.3.4 Communications Interfaces

- 1.The major communication for location purposes will be done by Google Api,the data is accessed by the Google by using the Google Api's.
- 2.The interface between the firebase DB and the system will be done by using http protocol

4.4 Nonfunctional Requirements

4.4.1 Performance Requirements

- 1. Statistical Analysis
The performance for this feature depends on the usage of the Smart Crime Reporter App. An analysis is done over the usage of the battery consumption on the phone.
- 2. Authentication
The performance for this feature is mainly dependent on the User Identity as the user need to fill up all the Demographic details while login into the App. In order to verify the user's validity we will use their Contact numbers to sign up into the App so that we can track the User if they try to Misuse.

4.4.2 Safety Requirements

- The identity of the User should be maintained in the database such as their Address and Contact numbers so as to track them if the App is not used properly.
- The database should be periodically maintained and have to keep upon it
- The data which is updated by the user should be committed in the database.

4.4.3 Security Requirements

- The major security requirements for the system will be the safeguarding of the user data from any kind of exploit.
- In order to protect the user identity and authentication it will be encoded in the QR Code which only the QR Scanner will be able to scan.

Chapter 5

System Design

5.1 System Requirements Definition

Our system is an Android application on a mobile device, that functions according to the emergency occurring at any place, the App will help the User to report about crime instantly. Our System mainly focuses on Voice Recognition, at any emergency the App triggers as soon as Keywords are spoken, the App also include Image Detection technology that captures the images in surrounding of a crime scene, through the use of GPS location the exact location is tracked and same is forwarded to the Police Side. The voice call Api will help to place a call at the Police end and things will be informed.

5.1.1 Functional Requirements

1. The customers must register for create the account and Login using Username and Password to use our Application.
2. The User Can also view Safety Tips about Particular Crimes
3. User can also detect Image at Crime spot using their Phones Camera .

Use-case Diagram

The figure illustrates the Use Case diagram of the system. In general, A Use Case diagram in its simplest, is a representation of a user's interaction with the system that shows the relationship between the users and the different use cases in which the user is involved. Use case diagram is use to show the overall flow of the system. The User must Login so as to Report Crimes they can also Read Safety tips regarding Crimes occurring in surrounding. The Location of crime Spot is also calculated with the help of GPS Locator.

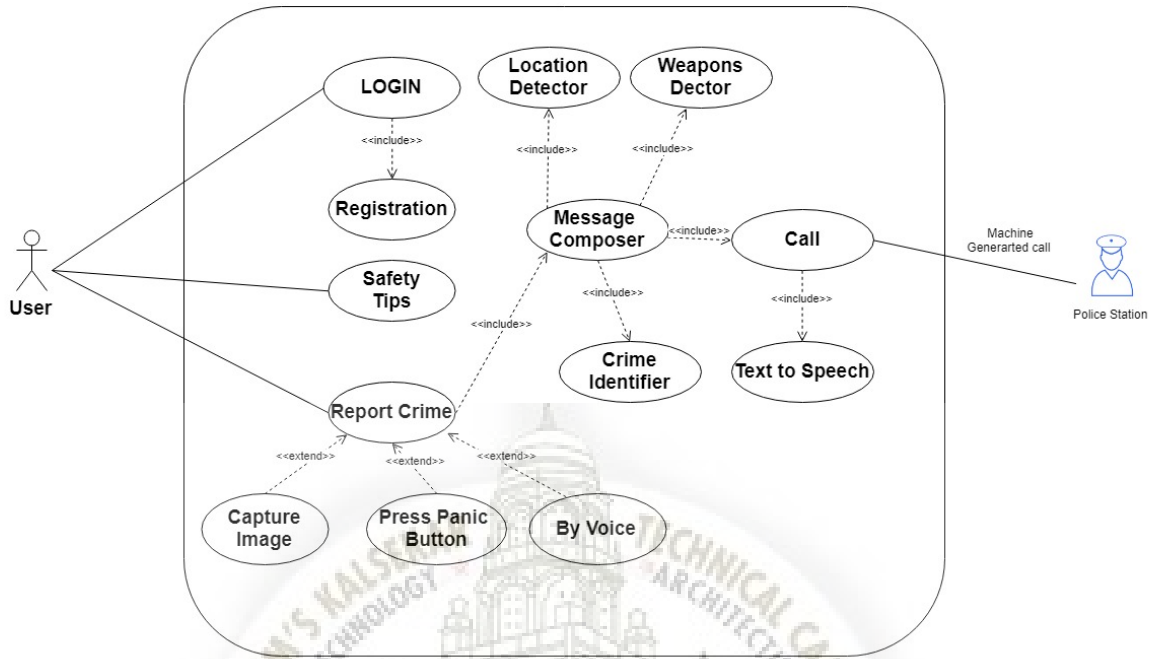


Figure 5.1: Use case of Smart Crime Reporter

Data-flow Diagram

The figure above illustrates the Data-Flow Diagram of the system. A Data-Flow Diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated. DFDs' can also be used for the visualization of data processing.

The first diagram is the level 0 Data flow diagram of Smart Crime Reporter the system in which, the modules of User and Police will be there after the deployment are shown. DFD Level 0 It contains total number of process(1) in our DFD level 0 diagram.It has User and Police.

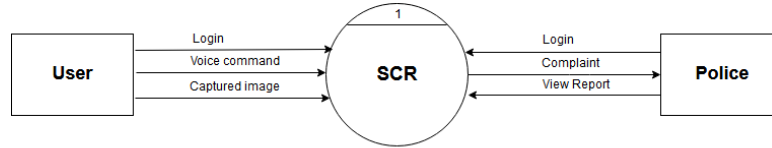


Fig:DFD level 0

Figure 5.2: Data flow Diagram Level 0

The second diagram is the level 1 Data Flow Diagram of the system in which the modules are briefly explained with their functionalities.

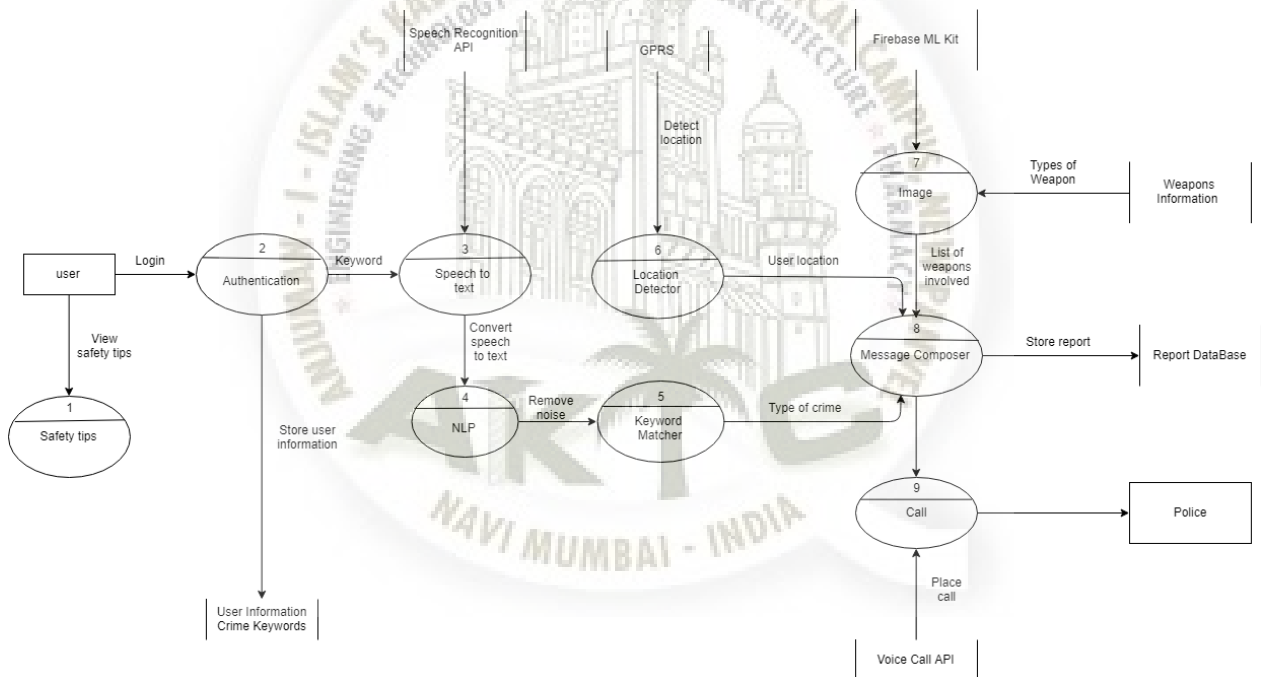


Figure 5.3: Data flow Diagram Level 1

The Third diagram is the level 1 Data Flow Diagram of the system in which the modules are briefly explained with their functionalities.

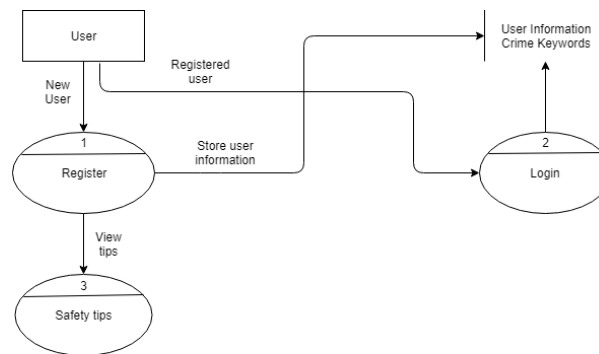


Fig: DFD 2 - Authentication

Figure 5.4: Data flow Diagram Level 2a

The Third diagram is the level 1 Data Flow Diagram of the system in which the modules are briefly explained with their functionalities.

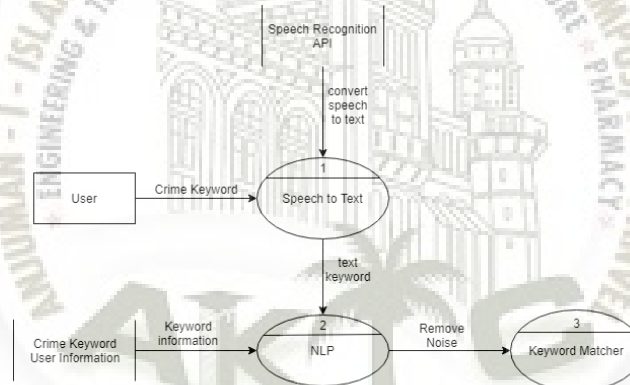


Fig: DFD 2 - Speech Recognition

Figure 5.5: Data flow Diagram Level 2b

5.1.2 System requirements (non-functional requirements)

The system should have high performance and low failure rates. The hardware and software should be able to transmit and receive data from the servers. The requests mainly depends upon the availability of the components in the inventory. For requesting a specific component, User must be authenticated first and then authorized to request for that particular equipment.

5.1.3 Safety Requirements

To provide the system from unauthorized user we are using QR Code for authentication and as a basic requirement to validate the user which will provide a better safety feature .The database of the system is done using Firebase which is ease and also safe to implement.

5.1.4 Security Requirements

The major security requirement for the system must be the safeguarding of the user data from any kind of exploit. Before any user want to access the system, they are required to input Username and password. Each password must be between 8-12 characters.

Database Schema/ E-R Diagram

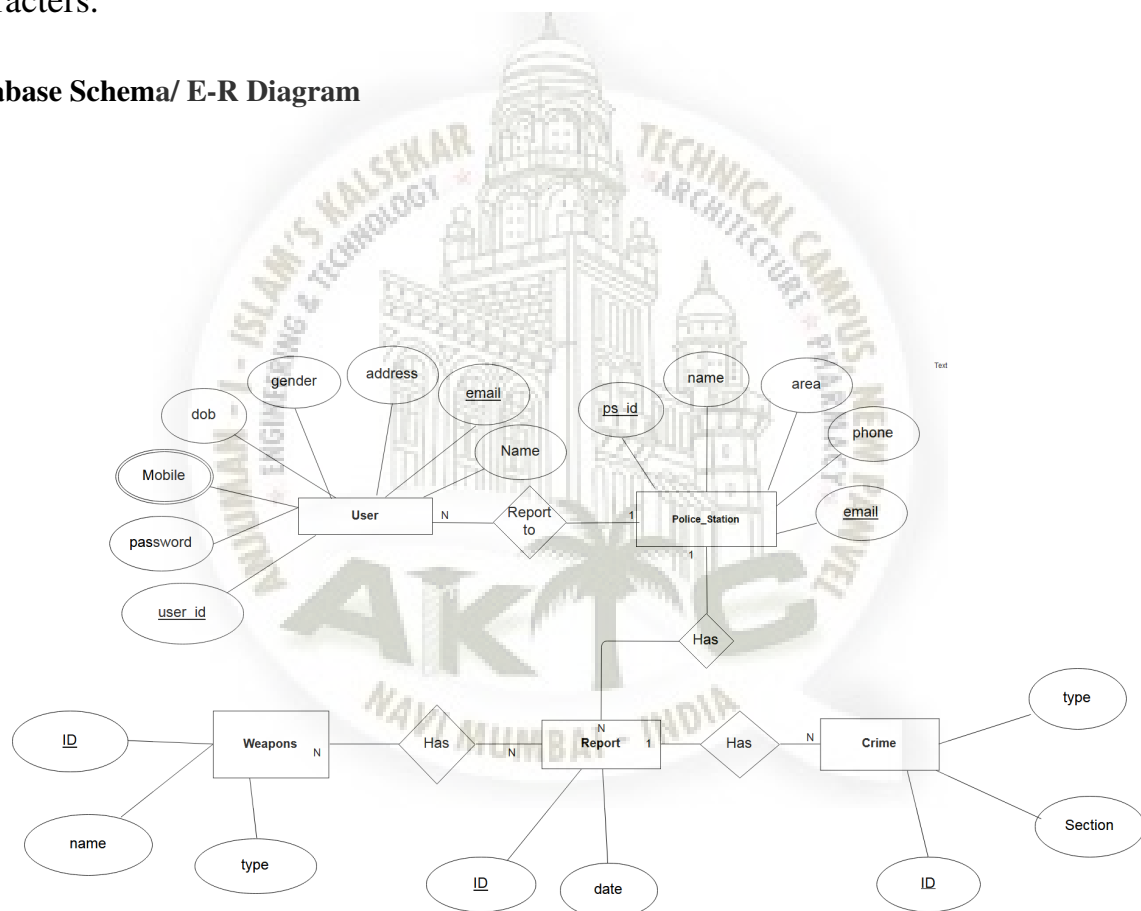


Figure 5.6: Entity-Relationship Diagram

5.2 System Architecture Design

Figure below is the System Architecture of the project. A system architecture or systems architecture is the conceptual model that defines the structure, behavior,

and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system.

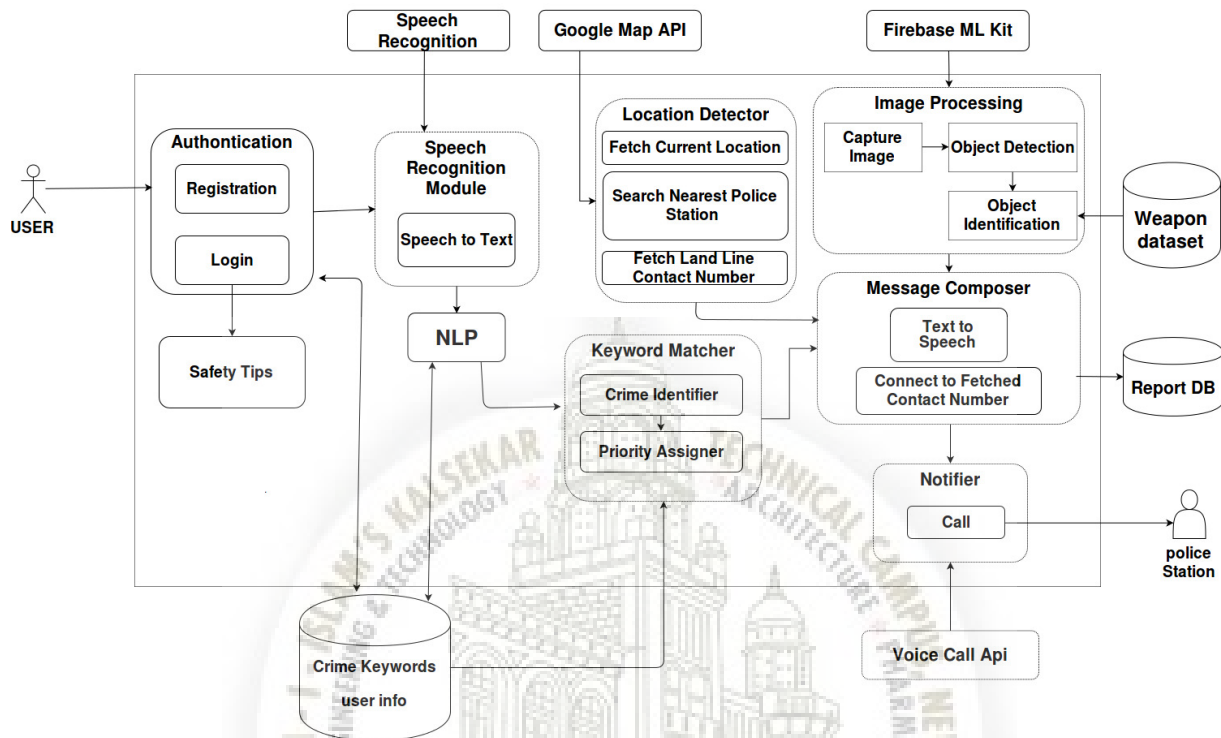


Figure 5.7: System Architecture of Smart Crime Reporter

5.3 Sub-system Development

A modular description provides detailed information about the module and its supported components which is accessible in different manner.

5.3.1 Speech Recognition

It is one of the most important module of our App, where Speech is captured in terms of Keywords these words are already defined in our dataset as soon as User speaks the Keyword the App gets Triggered.

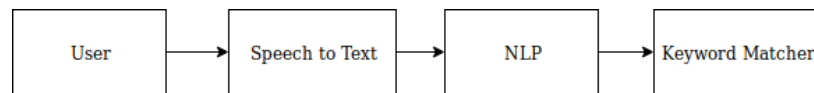


Figure 5.8: Modular Diagram of Speech Recognition

5.3.2 Keywords Matcher

The system uses NLP (Natural Language Processing) through this Keywords are matched, it depends on the User what keywords are spoken and accordingly it will be compared with the dataset already present.



Figure 5.9: Modular Diagram of Keywords matcher

5.3.3 Image Processing

This module mainly deals with the capturing of Images, it helps to capture the Images at the Crime scene it helps in detecting weapons and same is forwarded at the Police side, the whole things captures and it is send over the police side.



Figure 5.10: Modular Diagram of Image Processing

5.3.4 Location Detector

This Module is used to detect the location of the nearest police station. The google Map Api will help to calculate the latitude and Longitudes once the nearest Police station is fetched and same information is shared at the Police side.

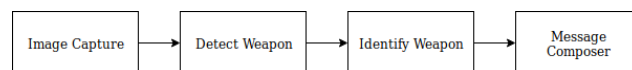


Figure 5.11: Modular Diagram of Location Detector

5.3.5 Report

Report is the actual Module which will report the crime, machines generated calls will be forwarded to the Police side whenever user reports a Crime.

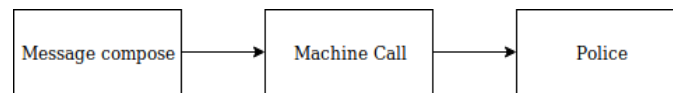


Figure 5.12: Modular Diagram of Report

5.4 Systems Integration

First module of our system is Login activity in Android. Second module is the QR Code which will be used for authentication and validation. Third is the statistical analysis which will generate an analysis of the usage of the cards at which merchant location and which outlet.

5.4.1 Class Diagram

This is the Class diagram of the system in which the modules which will be there after the deployment are shown the modules consist of Location, Safety Tips, Keywords, Reports which further extends with User, Police, Crime and Weapons it shows the relationship between them.

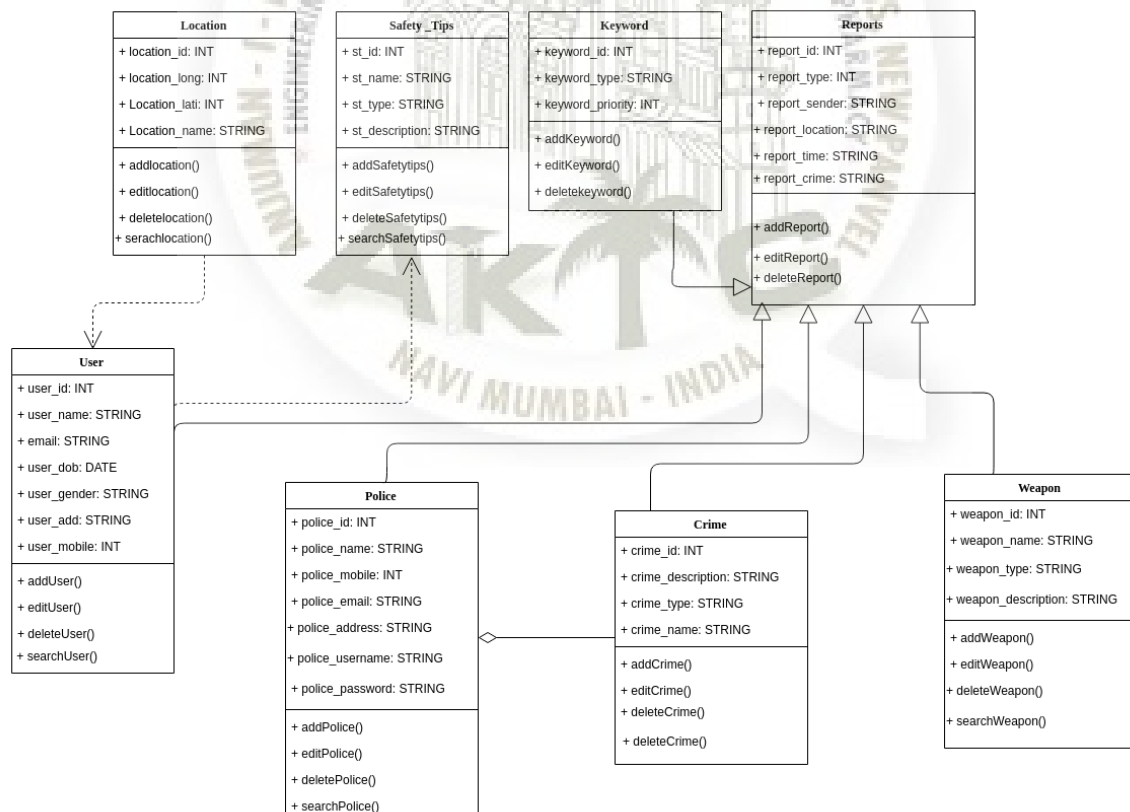


Figure 5.13: Class Diagram

5.4.2 Sequence Diagram

User

Here in User sequence diagram; the sequence is shown starting from Login into the App and then the User will view Safety tips, Speech recognition will record the Keywords spoken by the User through NLP the keywords are taken accurately if spelled incorrect. Location Detector will fetch the exact location at Crime scene and same will be forwarded at the Police Side.

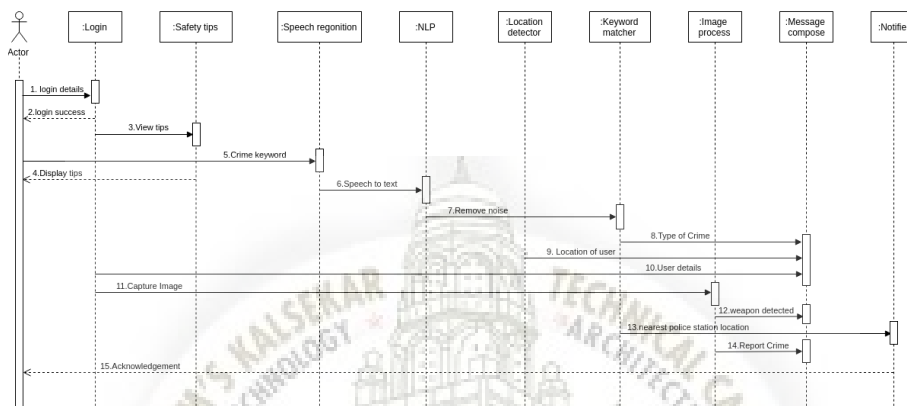


Figure 5.14: Sequence Diagram of User

Police

Here in Police sequence diagram; the sequence is shown starting from Login into the App and then the Police will view Safety tips, Check Report module will help the Police to view report of the Crimes. The police will also check the location of the Crime scene.

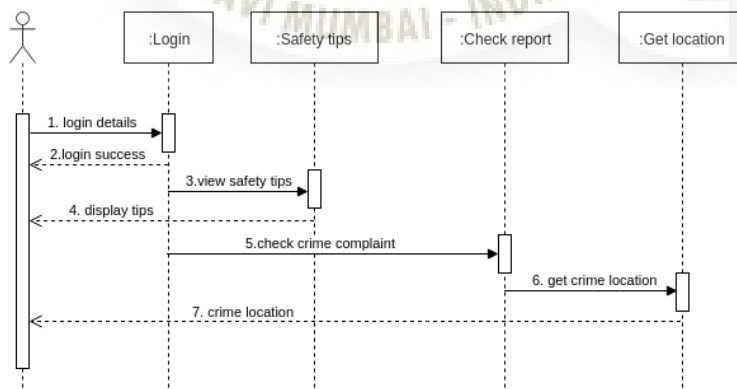


Figure 5.15: Sequence Diagram of Police

5.4.3 Component Diagram

The Component diagram basically describes the main highlighted components of the system. Here the main modules are Speech Recognition module, Google Map Api and Firebase ML kit. Component diagrams are used to visualize the organization and relationships among components in a system. These diagrams are also used to make executable systems.

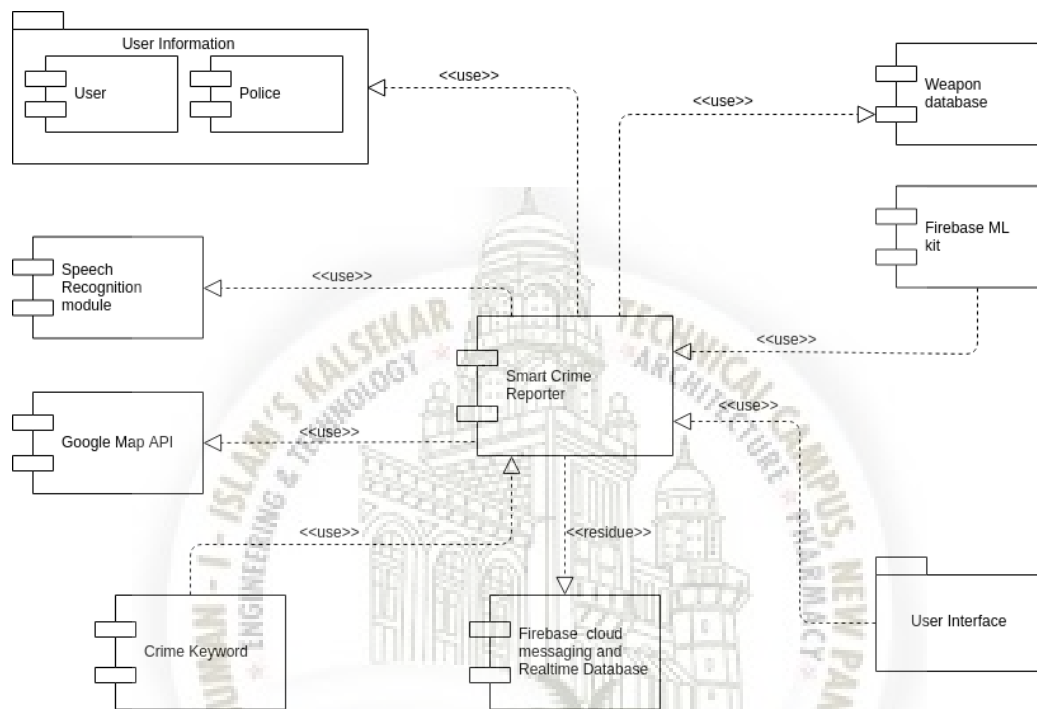


Figure 5.16: Component Diagram

- **Speech Recognition:**
The speech recognition will recognize the keywords which is stored in our database and it will be compared with the spoken Keywords.
- **Google Map Api:**
It will help to track the Exact location of the crime spot so the Police can reach accurately at Crime Scene.
- **firebase ML Kit:**
It is used to detect object in the crime scene which will help the Cops to Know the situation at crime scene that the culprits have what Weapons with them.

5.4.4 Deployment Diagram

Deployment diagram is a structure diagram which shows architecture of the system as deployment of software artifacts to deployment targets. Here the main modules are Firebase ML Kit, Weapons Dataset, Location detector Image Processing.

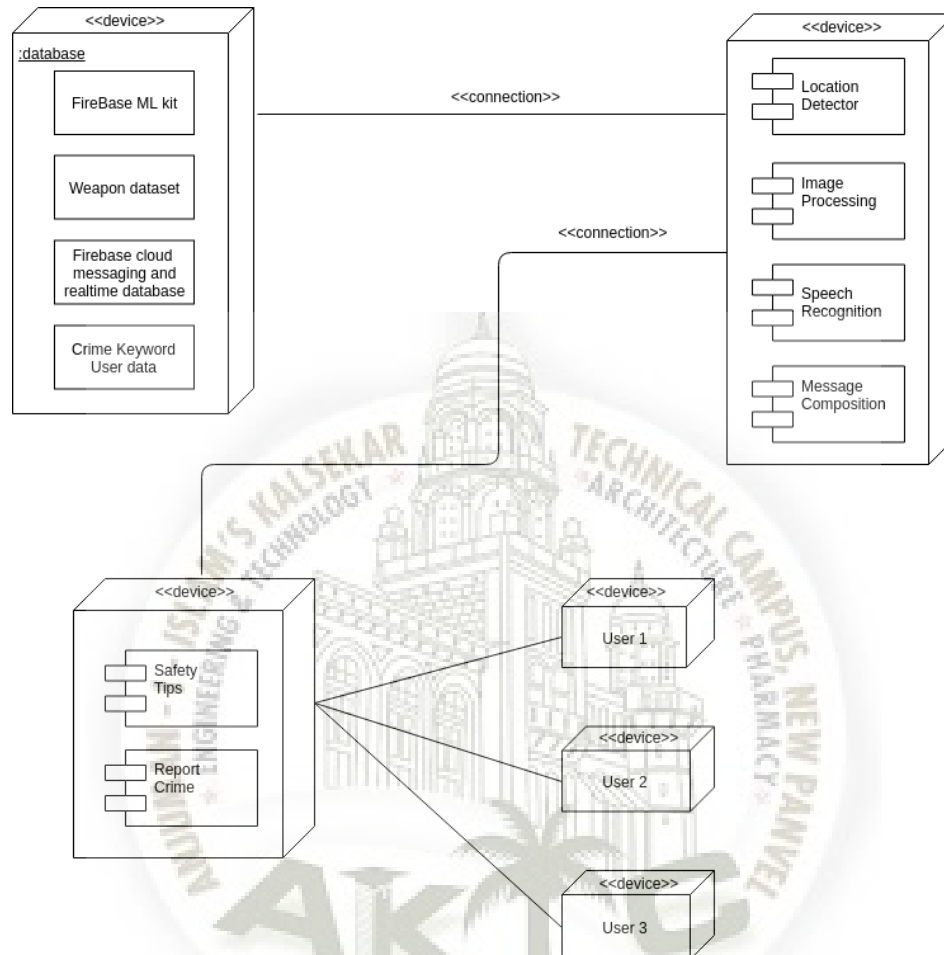


Figure 5.17: Deployment Diagram

- **Firebase ML Kit:**

It is used to detect object in the crime scene which will help the Cops to Know the situation at crime scene that the culprits have what Weapons with them.

- **Weapon Dataset:**

It has the data set of different weapons which are used in Crime, our system will compare such weapons with the Crime scene so that Cops will have an idea about it.

- **Image Processing:** The camera feature will allow the App to capture Images at crime scene and those image will be Processed with Image processing and will be sent to the Police side.

Chapter 6

Implementation

6.1 Select User

This is the first view on Our App where it ask for Whether you are a user or Police



Figure 6.1: Select User

```
1 public class PreLoginActivity extends AppCompatActivity implements View.  
   OnClickListener {  
2  
3     @Override  
4     protected void onCreate(Bundle savedInstanceState) {  
5         super.onCreate(savedInstanceState);  
6         setContentView(R.layout.activity_pre_login);
```

```
7     findViewById(R.id.btn_user).setOnClickListener(this);
8     findViewById(R.id.btn_police).setOnClickListener(this);
9 }
10
11 @Override
12 public void onClick(View v) {
13     switch (v.getId()) {
14         case R.id.btn_user:
15
16             Intent i = new Intent(PreLoginActivity.this, LoginActivity.class);
17             startActivity(i);
18             // finish();
19             break;
20         case R.id.btn_police:
21             Intent intent = new Intent(PreLoginActivity.this,
22                                     PoliceLoginActivity.class);
23             startActivity(intent);
24
25             break;
26     }
27 }
```



6.2 Registration for User/Police

The User or police must have to register themselves in order to use the App, the registration mainly consist of details such as Name, Email id Contact Number.

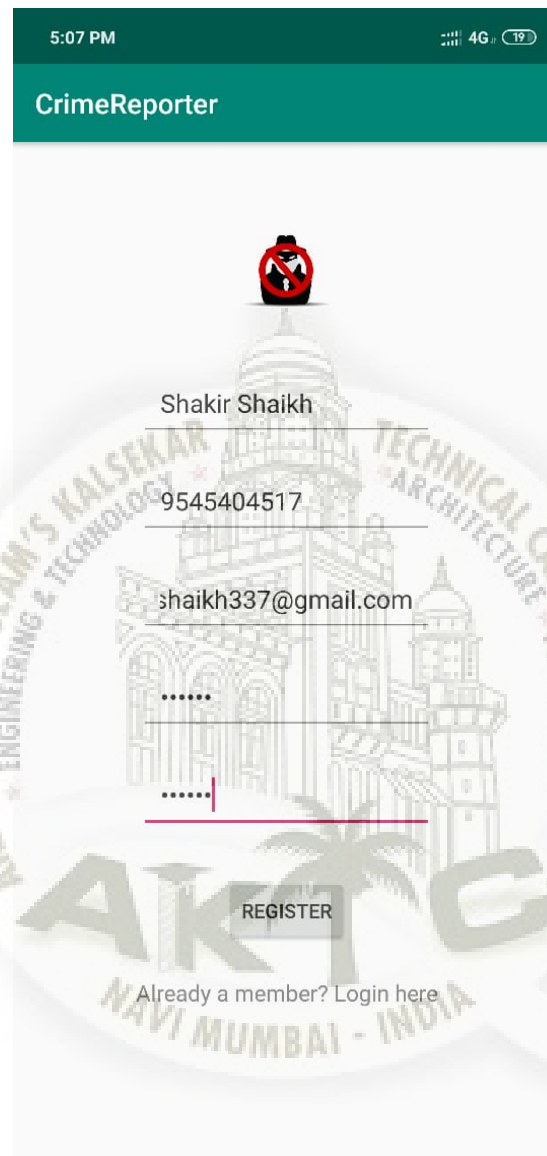


Figure 6.2: User Registration

```

1  private void registeruser () {
2      waitingDailog=new SpotsDialog.Builder().setContext(this)
3          .setMessage("Please Wait.....")
4          .setCancelable(false).build();
5      final String email=edittextEmail.getText().toString().trim();
6      String password=edittextPassword.getText().toString().trim();
7      String password2=edittextpassword2.getText().toString().trim();
8      final String name=editTextName.getText().toString().trim();
9      final String phone=editTextPhone.getText().toString().trim();
10     final String type="Police";
11     if(name.isEmpty()){
12         editTextName.setError("Name is required");

```

```

13         editTextName.requestFocus();
14         return;
15     }
16     if(phone.isEmpty()){
17         editTextPhone.setError("Email is required");
18         editTextPhone.requestFocus();
19         return;
20     }
21     if(email.isEmpty()){
22         edittextEmail.setError("Email is required");
23         edittextEmail.requestFocus();
24         return;
25     }
26     if (!Patterns.EMAIL_ADDRESS.matcher(email).matches()){
27         edittextEmail.setError("Please Enter the vaid email");
28         edittextEmail.requestFocus();
29         return;
30     }
31     if(password.isEmpty()){
32         edittextPassword.setError("Password is required");
33         edittextPassword.requestFocus();
34         return;
35     }
36     if(password.length()<6){
37         edittextPassword.setError("Password must have at least 6 character")
38         ;
39         edittextPassword.requestFocus();
40         return;
41     }
42     if(!password.equals(password2)){
43         edittextpassword2.setError("Password does not match");
44         edittextpassword2.requestFocus();
45         return;
46     }
47     waitingDailog.show();
48     mAuth.createUserWithEmailAndPassword(email, password).
49         addOnCompleteListener(new OnCompleteListener<AuthResult>() {
50     @Override
51     public void onComplete(@NonNull Task<AuthResult> task) {
52         if(task.isSuccessful()){
53             User user=new User(name, email, phone, type);
54             final Location location=new Location(latitude, longitude);
55             Toast.makeText(PoliceSingupActivity.this, mAuth.getUid(),
56                 Toast.LENGTH_SHORT).show();
57             mDatabase.child("users").child(mAuth.getUid()).setValue(user)
58             .addOnCompleteListener(new OnCompleteListener<Void>() {
59     @Override
60     public void onComplete(@NonNull Task<Void> task) {
61             if (task.isSuccessful()){
62                 mDatabase.child("police").child(mAuth.getUid()).setValue(location)
63                 .addOnCompleteListener(new OnCompleteListener<Void>() {
64     @Override
65     public void onComplete(@NonNull Task<Void> task) {
66             if (task.isSuccessful()) {
67                 Toast.makeText(getApplicationContext(), "Successfully registered",
68                     Toast.LENGTH_LONG).show();
69                 Intent intent=new
70                     Intent(PoliceSingupActivity.this, PoliceLoginActivity.class);

```

```
69         intent.addFlags( Intent.  
70             FLAG_ACTIVITY_NEW_TASK | Intent.FLAG_ACTIVITY_CLEAR_TASK);  
71     startActivity( intent );  
72  
73     finish ();  
74     waitingDailog.dismiss ();  
75     }  
76     else {  
77     Toast.makeText( getApplicationContext () , "location Failed" , Toast.  
78         LENGTHLONG) .show ();  
79     waitingDailog.dismiss ();  
80     }  
81     }  
82     }  
83     }  
84     }  
85     }  
86     }  
87     } else {  
88     Toast.makeText( getApplicationContext () , "Email Are already  
89         registered" , Toast.LENGTH.SHORT) .show ();  
90     waitingDailog.dismiss ();  
91     }  
92     }  
93     }  
94     }  
95     }  
96     }  
97     }  
98     }  
99     }  
100     }  
101     }  
102     }  
103     }  
104     }  
105     }  
106     }  
107     }  
108     }  
109     }  
110     }  
111     }  
112     }  
113     }  
114     }  
115     }  
116     }  
117     }  
118     }  
119     }  
120     }  
121     }  
122     }  
123     }  
124     }  
125     }  
126     }  
127     }  
128     }  
129     }  
130     }  
131     }  
132     }  
133     }  
134     }  
135     }  
136     }  
137     }  
138     }  
139     }  
140     }  
141     }  
142     }  
143     }  
144     }  
145     }  
146     }  
147     }  
148     }  
149     }  
150     }  
151     }  
152     }  
153     }  
154     }  
155     }  
156     }  
157     }  
158     }  
159     }  
160     }  
161     }  
162     }  
163     }  
164     }  
165     }  
166     }  
167     }  
168     }  
169     }  
170     }  
171     }  
172     }  
173     }  
174     }  
175     }  
176     }  
177     }  
178     }  
179     }  
180     }  
181     }  
182     }  
183     }  
184     }  
185     }  
186     }  
187     }  
188     }  
189     }  
190     }  
191     }  
192     }  
193     }  
194     }  
195     }  
196     }  
197     }  
198     }  
199     }  
200     }  
201     }  
202     }  
203     }  
204     }  
205     }  
206     }  
207     }  
208     }  
209     }  
210     }  
211     }  
212     }  
213     }  
214     }  
215     }  
216     }  
217     }  
218     }  
219     }  
220     }  
221     }  
222     }  
223     }  
224     }  
225     }  
226     }  
227     }  
228     }  
229     }  
230     }  
231     }  
232     }  
233     }  
234     }  
235     }  
236     }  
237     }  
238     }  
239     }  
240     }  
241     }  
242     }  
243     }  
244     }  
245     }  
246     }  
247     }  
248     }  
249     }  
250     }  
251     }  
252     }  
253     }  
254     }  
255     }  
256     }  
257     }  
258     }  
259     }  
260     }  
261     }  
262     }  
263     }  
264     }  
265     }  
266     }  
267     }  
268     }  
269     }  
270     }  
271     }  
272     }  
273     }  
274     }  
275     }  
276     }  
277     }  
278     }  
279     }  
280     }  
281     }  
282     }  
283     }  
284     }  
285     }  
286     }  
287     }  
288     }  
289     }  
290     }  
291     }  
292     }  
293     }  
294     }  
295     }  
296     }  
297     }  
298     }  
299     }  
300     }  
301     }  
302     }  
303     }  
304     }  
305     }  
306     }  
307     }  
308     }  
309     }  
310     }  
311     }  
312     }  
313     }  
314     }  
315     }  
316     }  
317     }  
318     }  
319     }  
320     }  
321     }  
322     }  
323     }  
324     }  
325     }  
326     }  
327     }  
328     }  
329     }  
330     }  
331     }  
332     }  
333     }  
334     }  
335     }  
336     }  
337     }  
338     }  
339     }  
340     }  
341     }  
342     }  
343     }  
344     }  
345     }  
346     }  
347     }  
348     }  
349     }  
350     }  
351     }  
352     }  
353     }  
354     }  
355     }  
356     }  
357     }  
358     }  
359     }  
360     }  
361     }  
362     }  
363     }  
364     }  
365     }  
366     }  
367     }  
368     }  
369     }  
370     }  
371     }  
372     }  
373     }  
374     }  
375     }  
376     }  
377     }  
378     }  
379     }  
380     }  
381     }  
382     }  
383     }  
384     }  
385     }  
386     }  
387     }  
388     }  
389     }  
390     }  
391     }  
392     }  
393     }  
394     }  
395     }  
396     }  
397     }  
398     }  
399     }  
400     }  
401     }  
402     }  
403     }  
404     }  
405     }  
406     }  
407     }  
408     }  
409     }  
410     }  
411     }  
412     }  
413     }  
414     }  
415     }  
416     }  
417     }  
418     }  
419     }  
420     }  
421     }  
422     }  
423     }  
424     }  
425     }  
426     }  
427     }  
428     }  
429     }  
430     }  
431     }  
432     }  
433     }  
434     }  
435     }  
436     }  
437     }  
438     }  
439     }  
440     }  
441     }  
442     }  
443     }  
444     }  
445     }  
446     }  
447     }  
448     }  
449     }  
450     }  
451     }  
452     }  
453     }  
454     }  
455     }  
456     }  
457     }  
458     }  
459     }  
460     }  
461     }  
462     }  
463     }  
464     }  
465     }  
466     }  
467     }  
468     }  
469     }  
470     }  
471     }  
472     }  
473     }  
474     }  
475     }  
476     }  
477     }  
478     }  
479     }  
480     }  
481     }  
482     }  
483     }  
484     }  
485     }  
486     }  
487     }  
488     }  
489     }  
490     }  
491     }  
492     }  
493     }  
494     }  
495     }  
496     }  
497     }  
498     }  
499     }  
500     }  
501     }  
502     }  
503     }  
504     }  
505     }  
506     }  
507     }  
508     }  
509     }  
510     }  
511     }  
512     }  
513     }  
514     }  
515     }  
516     }  
517     }  
518     }  
519     }  
520     }  
521     }  
522     }  
523     }  
524     }  
525     }  
526     }  
527     }  
528     }  
529     }  
530     }  
531     }  
532     }  
533     }  
534     }  
535     }  
536     }  
537     }  
538     }  
539     }  
540     }  
541     }  
542     }  
543     }  
544     }  
545     }  
546     }  
547     }  
548     }  
549     }  
550     }  
551     }  
552     }  
553     }  
554     }  
555     }  
556     }  
557     }  
558     }  
559     }  
560     }  
561     }  
562     }  
563     }  
564     }  
565     }  
566     }  
567     }  
568     }  
569     }  
570     }  
571     }  
572     }  
573     }  
574     }  
575     }  
576     }  
577     }  
578     }  
579     }  
580     }  
581     }  
582     }  
583     }  
584     }  
585     }  
586     }  
587     }  
588     }  
589     }  
590     }  
591     }  
592     }  
593     }  
594     }  
595     }  
596     }  
597     }  
598     }  
599     }  
600     }  
601     }  
602     }  
603     }  
604     }  
605     }  
606     }  
607     }  
608     }  
609     }  
610     }  
611     }  
612     }  
613     }  
614     }  
615     }  
616     }  
617     }  
618     }  
619     }  
620     }  
621     }  
622     }  
623     }  
624     }  
625     }  
626     }  
627     }  
628     }  
629     }  
630     }  
631     }  
632     }  
633     }  
634     }  
635     }  
636     }  
637     }  
638     }  
639     }  
640     }  
641     }  
642     }  
643     }  
644     }  
645     }  
646     }  
647     }  
648     }  
649     }  
650     }  
651     }  
652     }  
653     }  
654     }  
655     }  
656     }  
657     }  
658     }  
659     }  
660     }  
661     }  
662     }  
663     }  
664     }  
665     }  
666     }  
667     }  
668     }  
669     }  
670     }  
671     }  
672     }  
673     }  
674     }  
675     }  
676     }  
677     }  
678     }  
679     }  
680     }  
681     }  
682     }  
683     }  
684     }  
685     }  
686     }  
687     }  
688     }  
689     }  
690     }  
691     }  
692     }  
693     }  
694     }  
695     }  
696     }  
697     }  
698     }  
699     }  
700     }  
701     }  
702     }  
703     }  
704     }  
705     }  
706     }  
707     }  
708     }  
709     }  
710     }  
711     }  
712     }  
713     }  
714     }  
715     }  
716     }  
717     }  
718     }  
719     }  
720     }  
721     }  
722     }  
723     }  
724     }  
725     }  
726     }  
727     }  
728     }  
729     }  
730     }  
731     }  
732     }  
733     }  
734     }  
735     }  
736     }  
737     }  
738     }  
739     }  
740     }  
741     }  
742     }  
743     }  
744     }  
745     }  
746     }  
747     }  
748     }  
749     }  
750     }  
751     }  
752     }  
753     }  
754     }  
755     }  
756     }  
757     }  
758     }  
759     }  
760     }  
761     }  
762     }  
763     }  
764     }  
765     }  
766     }  
767     }  
768     }  
769     }  
770     }  
771     }  
772     }  
773     }  
774     }  
775     }  
776     }  
777     }  
778     }  
779     }  
780     }  
781     }  
782     }  
783     }  
784     }  
785     }  
786     }  
787     }  
788     }  
789     }  
790     }  
791     }  
792     }  
793     }  
794     }  
795     }  
796     }  
797     }  
798     }  
799     }  
800     }  
801     }  
802     }  
803     }  
804     }  
805     }  
806     }  
807     }  
808     }  
809     }  
810     }  
811     }  
812     }  
813     }  
814     }  
815     }  
816     }  
817     }  
818     }  
819     }  
820     }  
821     }  
822     }  
823     }  
824     }  
825     }  
826     }  
827     }  
828     }  
829     }  
830     }  
831     }  
832     }  
833     }  
834     }  
835     }  
836     }  
837     }  
838     }  
839     }  
840     }  
841     }  
842     }  
843     }  
844     }  
845     }  
846     }  
847     }  
848     }  
849     }  
850     }  
851     }  
852     }  
853     }  
854     }  
855     }  
856     }  
857     }  
858     }  
859     }  
860     }  
861     }  
862     }  
863     }  
864     }  
865     }  
866     }  
867     }  
868     }  
869     }  
870     }  
871     }  
872     }  
873     }  
874     }  
875     }  
876     }  
877     }  
878     }  
879     }  
880     }  
881     }  
882     }  
883     }  
884     }  
885     }  
886     }  
887     }  
888     }  
889     }  
890     }  
891     }  
892     }  
893     }  
894     }  
895     }  
896     }  
897     }  
898     }  
899     }  
900     }  
901     }  
902     }  
903     }  
904     }  
905     }  
906     }  
907     }  
908     }  
909     }  
910     }  
911     }  
912     }  
913     }  
914     }  
915     }  
916     }  
917     }  
918     }  
919     }  
920     }  
921     }  
922     }  
923     }  
924     }  
925     }  
926     }  
927     }  
928     }  
929     }  
930     }  
931     }  
932     }  
933     }  
934     }  
935     }  
936     }  
937     }  
938     }  
939     }  
940     }  
941     }  
942     }  
943     }  
944     }  
945     }  
946     }  
947     }  
948     }  
949     }  
950     }  
951     }  
952     }  
953     }  
954     }  
955     }  
956     }  
957     }  
958     }  
959     }  
960     }  
961     }  
962     }  
963     }  
964     }  
965     }  
966     }  
967     }  
968     }  
969     }  
970     }  
971     }  
972     }  
973     }  
974     }  
975     }  
976     }  
977     }  
978     }  
979     }  
980     }  
981     }  
982     }  
983     }  
984     }  
985     }  
986     }  
987     }  
988     }  
989     }  
990     }  
991     }  
992     }  
993     }  
994     }  
995     }  
996     }  
997     }  
998     }  
999     }  
1000    }
```

6.3 Login for User/Police

The user will see this Screen once he wants to login into the App, The User must enter correct Credentials in order to login Successfully.



Figure 6.3: User Login

```

1  mAuth.signInWithEmailAndPassword(email, password)
2      .addOnCompleteListener(this, new OnCompleteListener<AuthResult>
3      >() {
4      @Override
5      public void onComplete(@NonNull Task<AuthResult> task) {
6      if (task.isSuccessful()) {
7      FirebaseDatabase.getInstance().getReference()
8      .child("users").child(mAuth.getCurrentUser().getUid()).
9      child("type");
10     FirebaseDatabase.addValueEventListener(new ValueEventListener() {
11     @Override
12     public void onDataChange(@NonNull DataSnapshot dataSnapshot) {

```

```
11     String type=dataSnapshot.getValue().toString();
12     Toast.makeText(getApplicationContext(),type,Toast.LENGTHLONG).show();
13     if (type.equals("user")){
14         SharedPreferences sp=getSharedPreferences("login",
15             MODE.PRIVATE);
16         SharedPreferences.Editor edit=sp.edit();
17         edit.putString("email",email);
18         edit.putString("password",password);
19         edit.putString("type","User");
20         edit.commit();
21
22         startActivity(new Intent(LoginActivity.this,
23             HomeActivity.class));
24         finish();
25         waitingDailog.dismiss();
26     }else {
27         waitingDailog.dismiss();
28     }
29
30     }
31
32     @Override
33     public void onCancelled(@NonNull DatabaseError databaseError) {
34
35         }
36     });
37
38     } else {
39         // If sign in fails , display a message to the user.
40         Toast.makeText(getApplicationContext(),"Login Failed
41             ", Toast.LENGTH.SHORT).show();
42         waitingDailog.dismiss();
43     }
44 }
45
46 });
47
48 }
```

6.4 User Panel



Figure 6.4: {User Panel

```

1  @Override
2  public boolean onOptionsItemSelected(MenuItems item) {
3      switch (item.getItemId()) {
4          case R.id.logout:
5              SharedPreferences sharedPreferences=
6                  sharedPreferences.getSharedPreferences("login",MODE_PRIVATE);
7              SharedPreferences.Editor editor=sharedPreferences.edit();
8              editor.clear();
9              editor.apply();
10             FirebaseAuth.getInstance().signOut();
11             startActivity(new Intent(this, MainActivity.class));
12             finish();
13             return true;
14         case R.id.profile:
15             Toast.makeText(this, "Profile Module Not Create", Toast.LENGTH_SHORT).show();
16             return true;

```

```
16     case R.id.settings :
17         Toast.makeText(this, "Module Not Created", Toast.LENGTH.SHORT).
18             show();
19         return true;
20
21     default :
22         return super.onOptionsItemSelected(item);
23
24 }
```



6.5 Safety Tips



Figure 6.5: Safety Tips

```

1 class MyAdapter extends BaseExpandableListAdapter {
2     private Context ctx;
3     private HashMap<String , List<String>> ChildTitles;
4     private List<String> HeaderTitles;
5     MyAdapter(Context ctx , HashMap<String , List<String>> ChildTitles , List<String>
6         HeaderTitles){
7         this.ctx=ctx;
8         this.ChildTitles=ChildTitles;
9         this.HeaderTitles=HeaderTitles;
10    }
11
12    @Override
13    public int getGroupCount() {
14        return HeaderTitles.size();
15    }

```



```
16
17     @Override
18     public int getChildrenCount(int groupPosition) {
19         return ChildTitles.get(HeaderTitles.get(groupPosition)).size();
20     }
21
22     @Override
23     public Object getGroup(int groupPosition) {
24         return HeaderTitles.get(groupPosition);
25     }
26
27     @Override
28     public Object getChild(int groupPosition, int childPosition) {
29         return ChildTitles.get(HeaderTitles.get(groupPosition)).get(
30             childPosition);
31     }
32
33     @Override
34     public long getGroupId(int groupPosition) {
35         return groupPosition;
36     }
37
38     @Override
39     public long getChildId(int groupPosition, int childPosition) {
40         return groupPosition;
41     }
42
43     @Override
44     public boolean hasStableIds() {
45         return false;
46     }
47
48     @Override
49     public View getGroupView(int groupPosition, boolean isExpanded, View
50         convertView, ViewGroup parent) {
51         String title=(String) this.getGroup(groupPosition);
52         if (convertView==null){
53             LayoutInflater inflater=(LayoutInflater) this.ctx.getSystemService(
54                 Context.LAYOUT_INFLATER_SERVICE);
55             convertView=inflater.inflate(R.layout.custom_header, null);
56         }
57         TextView txt=convertView.findViewById(R.id.idTitle);
58         txt.setTypeface(null, Typeface.BOLD);
59         txt.setText(title);
60
61         return convertView;
62     }
63 }
```

6.6 Report Activity

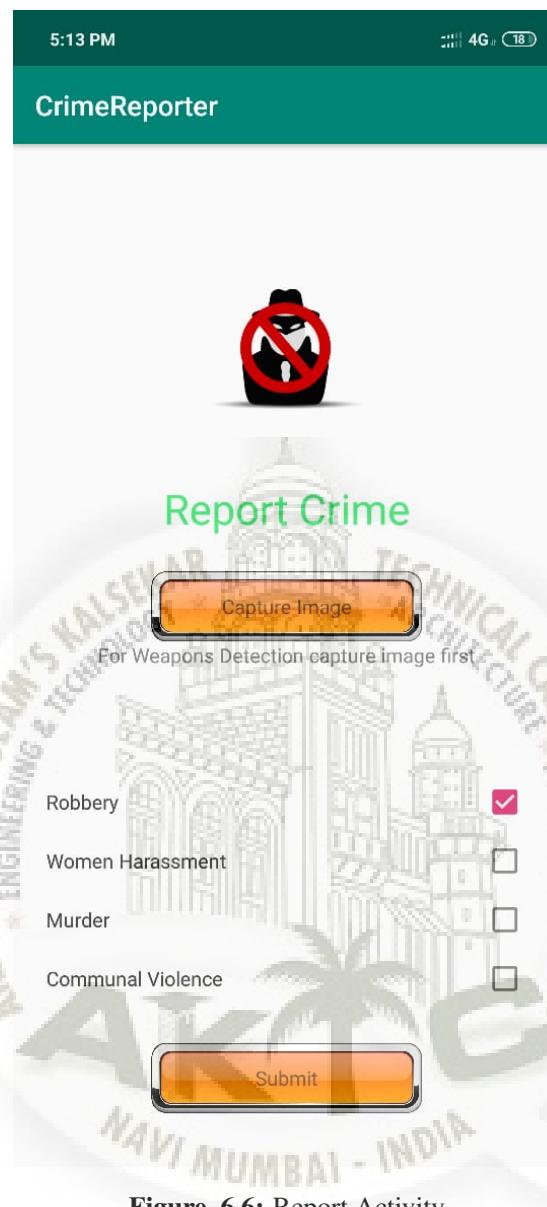


Figure 6.6: Report Activity

```

1 public class ReportActivity extends AppCompatActivity implements View.
   OnClickListener , LocationListener {
2     private TextView textView ,textView2;
3     private CheckedTextView c1 ,c2 ,c3 ,c4;
4     public String weapons;
5     public String objects ,policePhone;
6
7
8     private FirebaseAuth mAuth;
9
10    AlertDialog waitingDailog;
11    //new Code added for loaction
12    private static final int PERMISSIONS.REQUEST = 100;
13    private GoogleApiClient mGoogleApiClient;
14    LocationManager locationManager;
15    String policekey;

```

```

16  double latitude ;
17  double longitude ;
18  @Override
19  protected void onCreate(Bundle savedInstanceState) {
20      super.onCreate(savedInstanceState);
21      setContentView(R.layout.activity_report);
22      waitingDailog=new SpotsDialog.Builder().setContext(this)
23          .setMessage("Please Wait.....")
24          .setCancelable(false).build();
25      getLocation();
26      getNearestPolice();
27      if (ContextCompat.checkSelfPermission(getApplicationContext(), android.
28          Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.
29          PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(
30          getApplicationContext(), android.Manifest.permission.
31          ACCESS_COARSE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
32
33          ActivityCompat.requestPermissions(this, new String[]{android.
34              Manifest.permission.ACCESS_FINE_LOCATION, android.Manifest.
35              permission.ACCESS_COARSE_LOCATION}, 101);
36
37      }
38
39      textView=findViewById(R.id.object);
40      mAuth=FirebaseAuth.getInstance();
41      textView2=findViewById(R.id.weapon);
42      findViewById(R.id.capture_image).setOnClickListener(this);
43      findViewById(R.id.report_crime).setOnClickListener(this);
44      c1=findViewById(R.id.crime1);
45      c2=findViewById(R.id.crime2);
46      c3=findViewById(R.id.crime3);
47      c4=findViewById(R.id.crime4);
48      findViewById(R.id.crime1).setOnClickListener(this);
49      findViewById(R.id.crime2).setOnClickListener(this);
50      findViewById(R.id.crime3).setOnClickListener(this);
51      findViewById(R.id.crime4).setOnClickListener(this);
52
53      }
54
55      @Override
56      public void onClick(View v) {
57          switch(v.getId()){
58              case R.id.crime1:
59                  c1.toggle();
60                  break;
61              case R.id.crime2:
62                  c2.toggle();
63                  break;
64              case R.id.crime3:
65                  c3.toggle();
66                  break;
67              case R.id.crime4:
68                  c4.toggle();
69                  break;
70              case R.id.capture_image:
71                  getweapons();
72                  //startActivity(new Intent(this, WeaponsDetector.class));
73                  break;
74              case R.id.report_crime:

```

```

71         getLocation ();
72         report ();
73         break ;
74
75     }
76     // startActivity (new Intent (this , WeaponsDetector . class ));
77 }
78 public void getweapons () {
79     Intent intent=new Intent (getApplicationContext () , WeaponsDetector . class );
80     startActivityForResult (intent ,2);
81
82 }
83
84 @Override
85 protected void onActivityResult (int requestCode , int resultCode , @Nullable
86     Intent data) {
87     super .onActivityResult (requestCode , resultCode , data);
88     if (requestCode==2){
89         weapons=data .getStringExtra ("weapons");
90         objects=data .getStringExtra ("objects");
91         textView .setText ("Detected object :"+objects);
92         textView2 .setText ("Detected Weapons :"+weapons);
93
94         // Toast .makeText (this , data .getStringExtra ("weapons") , Toast .
95             LENGTHSHORT) .show ();
96     }
97
98
99 public void storeCrimeReport (String crimes){
100     if (latitude ==0){ getLocation ();}
101     waitingDailog .show ();
102     String title="This is Emergency Alert";
103     String body="Crime List :"+crimes+" Weapons :"+weapons;
104     Message message=new Message (title ,body , mAuth .getCurrentUser () .getUid () ,
105         policePhone , latitude , longitude);
106     DatabaseReference mReference= FirebaseDatabase .getInstance () .
107         getReference () .child ("messages");
108     mReference .push () .setValue (message)
109         .addOnCompleteListener (new OnCompleteListener <Void> () {
110             @Override
111             public void onComplete (@NonNull Task <Void> task) {
112                 if (task .isSuccessful ()) {
113                     Toast .makeText (ReportActivity .this , "Reported to
114                         nearest police station" , Toast .LENGTHLONG) .show
115                         ();
116                 }
117                 else {
118                     Toast .makeText (ReportActivity .this , "error generated
119                         " , Toast .LENGTHSHORT) .show ();
120                 }
121             }
122         });
123
124     waitingDailog .dismiss ();
125 }
126
127 public void getNearestPolice () {

```

```
125     waitingDialog.show();
126     final double[] endlatitude = new double[1];
127     final double[] endlongitude=new double[1];
128     final String[] policeKey = new String[1];
129     final float[] distance = {1000000000};
130     final float results []= new float[10];
131
132     DatabaseReference dbref=FirebaseDatabase.getInstance().getReference().
133     child("police");
134     dbref.addValueEventListener(new ValueEventListener() {
135         @Override
136         public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
137             for(DataSnapshot postSnapshot:dataSnapshot.getChildren()) {
138                 if(policeKey[0]==null)
139                     policeKey[0]=postSnapshot.getKey();
140                 endlatitude[0] = (double) postSnapshot.child("latitude").
141                 getValue();
142                 endlongitude[0] = (double) postSnapshot.child("longitude").
143                 getValue();
144                 Location.distanceBetween(latitude, longitude, endlatitude
145                 [0], endlongitude[0], results);
146
147                 if (distance[0] > results[0]) {
148                     distance[0] = results[0];
149                     Log.d("Dist",Float.toString(distance[0]));
150                     policeKey[0] = postSnapshot.getKey();
151                 }
152                 Log.d("results",Float.toString(results[0]));
153             }
154             Log.d("ffff", policeKey[0]);
155             searchPolicephone(policeKey[0]);
156         }
157     }
```

6.7 Weapon Detector



Figure 6.7: Weapon Detector

```

1 private void runDetector(Bitmap bitmap) {
2
3     final FirebaseVisionImage image=FirebaseVisionImage.fromBitmap(bitmap);
4     new InternetCheck(new InternetCheck.Consumer() {
5         @Override
6         public void accept(boolean internet) {
7
8             FirebaseVisionLabelDetectorOptions options=new
9                 FirebaseVisionLabelDetectorOptions.Builder()
10                    .setConfidenceThreshold(0.8f).build();
11             FirebaseVisionLabelDetector detector=FirebaseVision.
12                 getInstance().getVisionLabelDetector(options);
13             detector.detectInImage(image)
14                 .addOnSuccessListener(new OnSuccessListener<List<
15                     FirebaseVisionLabel>>() {
16                 @Override
17                 public void onSuccess(List<FirebaseVisionLabel>
18                     firebaseVisionLabels) {

```

```

15         processDataResult ( firebaseVisionLabels );
16
17     }
18 }
19     .addOnFailureListener ( new OnFailureListener () {
20         @Override
21         public void onFailure (@NonNull Exception e) {
22             Toast.makeText ( getApplicationContext () , "
23                 Error 2" , Toast.LENGTH_SHORT) . show () ;
24
25             Log.d ("EDMTERROR" , e.getMessage () ) ;
26             waitingDailog . dismiss () ;
27
28         }
29     } );
30
31 }
32 });
33
34 }
35 private void processDataResult ( List < FirebaseVisionLabel >
36     firebaseVisionLabels ) {
37
38
39     for ( FirebaseVisionLabel label : firebaseVisionLabels ) {
40         //checkWeapon ( label . getLabel () ) ;
41
42         object = label . getLabel () ;
43         checkWeapon ( object ) ;
44         Toast.makeText ( getApplicationContext () , "Device result : " + label .
45             getLabel () , Toast.LENGTH_SHORT) . show () ;
46
47     }
48     if ( waitingDailog . isShowing () )
49         waitingDailog . dismiss () ;
50     //cameraView . start () ;
51     //Intent intent = new Intent () ;
52     intent . putExtra ("objects" , object ) ;
53     setResult ( 2 , intent ) ;
54     finish () ;
55
56
57
58 }
59 public void checkWeapon ( String weapon ) {
60     // Toast.makeText ( this , "In checkWeapons method" , Toast.LENGTH_SHORT) .
61     show () ;
62     if ( weapon . equals ("Gun" ) ) {
63         weapons = weapon ;
64
65     } else if ( weapon . equals ("Nife" ) ) {
66         weapons = weapon ;
67     } else {
68         intent . putExtra ("weapons" , weapons ) ;
69     }
70 }

```

6.8 Police Panel



Figure 6.8: Police Panel

```

1 public class PolicePanelActivity extends AppCompatActivity {
2     public static String CHANNEL_ID="Its channel id";
3     private static String CHANNEL_NAME="Its channel Name ";
4     private static String CHANNEL_DESC="Its channel Description";
5     ImageView imageView;
6     FirebaseAuth mAuth;
7
8     @Override
9     protected void onCreate(Bundle savedInstanceState) {
10        super.onCreate(savedInstanceState);
11        setContentView(R.layout.activity_police_panel);
12        mAuth= FirebaseAuth.getInstance();
13        if (Build.VERSION.SDK_INT>=Build.VERSION_CODES.O){
14            NotificationChannel channel=new NotificationChannel(CHANNEL_ID,
15                CHANNEL_NAME, NotificationManager.IMPORTANCE.DEFAULT);
16            channel.setDescription(CHANNEL_DESC);
17            NotificationManager manager=getSystemService(NotificationManager.
18                class);

```



```
17         manager.createNotificationChannel(channel);
18     }
19
20     String authId=mAuth.getCurrentUser().getUid();
21
22     subscribeTO(authId);
23     findViewById(R.id.report_list).setOnClickListener(new View.
24         OnClickListener() {
25         @Override
26         public void onClick(View v) {
27
28             startActivity(new Intent(getApplicationContext(),ReportList.
29                 class));
30         }
31     });
32
33
34
35 }
36 private void subscribeTO(String authId){
37     FirebaseDatabase.getInstance().getReference().child("users").child(
38     authId)
39     .addListenerForSingleValueEvent(new ValueEventListener() {
40     @Override
41     public void onDataChange(@NonNull DataSnapshot dataSnapshot)
42     {
43         User user=dataSnapshot.getValue(User.class);
44         String mobile=user.phone;
45         FirebaseMessaging.getInstance().subscribeToTopic(mobile)
46         ;
47     }
48     @Override
49     public void onCancelled(@NonNull DatabaseError databaseError
50     ) {
51     }
52 }
53 });
54 }
```

6.9 Report List



Figure 6.9: Report List

```
1 private void setPhone(final String phone) {
2     mref=FirebaseDatabase.getInstance().getReference().child("messages");
3     mref.addValueEventListener(new ValueEventListener() {
4         @Override
5         public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
6             for(DataSnapshot dataSnapshot1 : dataSnapshot.getChildren()){
7                 message=dataSnapshot1.getValue(Message.class);
8                 if(message.police.equals(phone)){
9                     message=dataSnapshot1.getValue(Message.class);
10                    messages.add(message);
11                }
12            }
13        }
14        recyclerView.setAdapter(new ReportAdapter(messages,
15            getApplicationContext()));
16    }
17
18
19    @Override
20    public void onCancelled(@NonNull DatabaseError databaseError) {
21    }
22
23    });
24
25
26
27 }
```



6.10 Crime Location



Figure 6.10: Crime Location

```

1 @Override
2 public void onBindViewHolder(@NonNull ReportViewHolder reportViewHolder, int i)
3 {
4     //code for address finding
5     final double latitude=messages.get(i).latitude;
6     final double longitude=messages.get(i).logitude;
7     String add1 = null,add2,add3,add4;
8     final String labelLocation="Crime Location";
9     Geocoder geocoder = new Geocoder(context, Locale.getDefault());
10    try {
11        List<Address> addresses = geocoder.getFromLocation(latitude, longitude,
12            1);
13        add1=addresses.get(0).getAddressLine(0);
14    } catch (IOException e) {
15        e.printStackTrace();
16    }

```

```
17
18     final String title= messages.get(i).title;
19     String desc=messages.get(i).body;
20     reportViewHolder.title.setText(title);
21     reportViewHolder.desc.setText(desc);
22     reportViewHolder.add.setText(add1);
23     reportViewHolder.cardView.setOnClickListener(new View.OnClickListener() {
24         @Override
25         public void onClick(View v) {
26             Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("geo:<" +
27                 latitude + ">,<" + longitude + ">?q=<" + latitude + ">,<" +
28                 longitude + ">(" + labelLocation + ")"));
29             intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
30             context.startActivity(intent);
31         }
32     });
33 }
```



Chapter 7

System Testing

In This Chapter the System is being tested to find out the accuracy of the system. The Tested result is shown in the Table and the Result image is shown below. Testing is used to find the error rate and to find the loop holes in the System. It gives the clear idea about the working of the system and the problems in the System.

7.1 Test Cases and Test Results

Test ID	Test Case Title	Test Condition	System Behavior	Expected Result
T01	Login	Should be register user	Will connect to Firebase	Should login into system
T02	Object Detection	Should detect Images	Images are recognized	objects are Detected
T03	GPS Location	Search for Nearest police station	Exact Location are searched	Successfully fetching location.

Table 7.1: Test Cases Table

7.2 Sample of a Test Case

Title: Login Page – Authenticate Successfully on gmail.com

Description: A registered user should be able to successfully login at gmail.com.

Precondition: the user must already be registered with an email address and password.

Assumption: a supported browser is being used.

Test Steps:

1. Navigate to gmail.com
2. In the 'email' field, enter the email of the registered user.
3. Click the 'Next' button.
4. Enter the password of the registered user
5. Click 'Sign In'
6. If not registered click on Signup button

Expected Result: A page displaying the homepage should load, showing navigation menu at the top of the page.

Actual Result:

After successful login through Firebase, homepage is displayed saying Report Crime or view Safety Tips

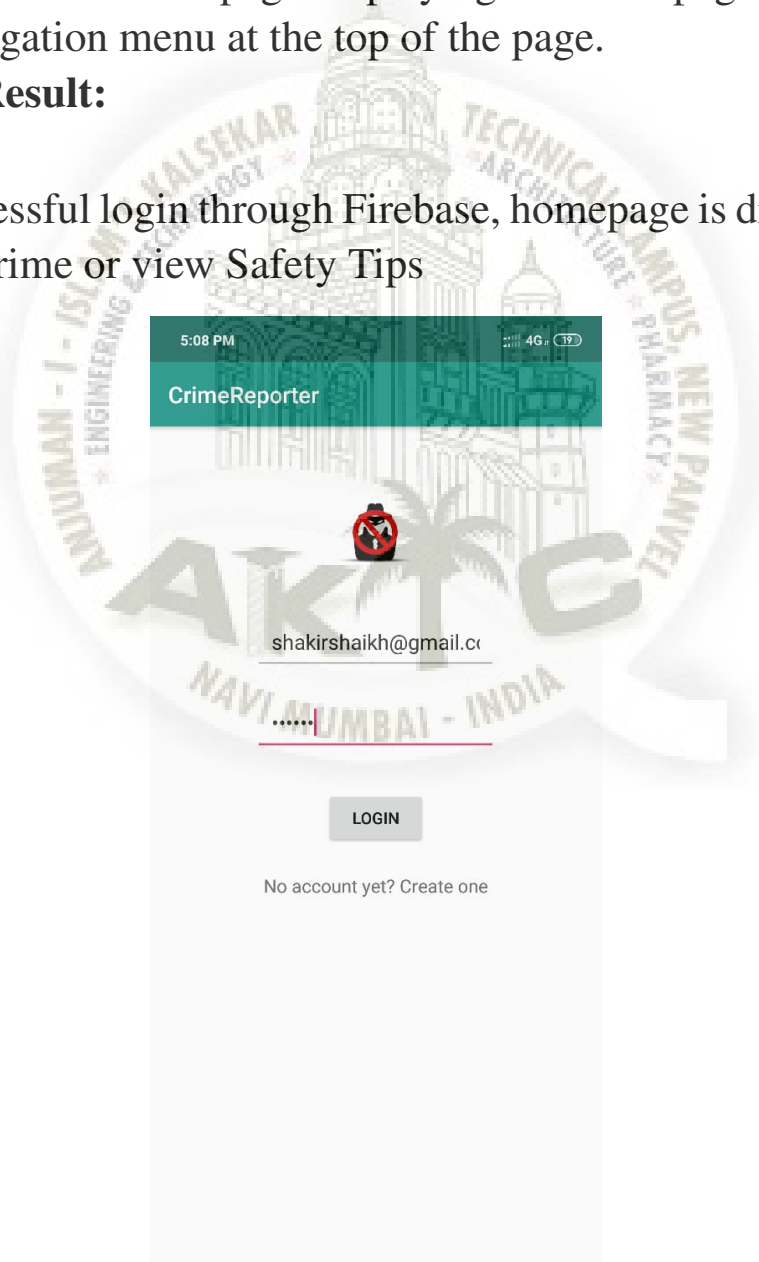


Figure 7.1: Login Page

Title:Image Detection

Description: Once the user login into the App he has options about reporting Crime and Detect Images, the Module is mainly used to detect Weapons at the Crime Scene.

Precondition: The user must already be registered with an email address and password.

Assumption: Images to be Captured

Test Steps:

1. Login with the registered username and password
2. App will view about reporting Crime option
3. There is an option of capturing images.
4. The Images will be Captured.

Expected Result: Images should be captured in the surrounding of Crime Scene

Actual Result:

The Images in the background are actually Captured viewing all the objects at the Crime Scene.



Figure 7.2: Object Detection

Title: GPS Location – Successfully fetching the location

Description: GPS location will help to fetch the location at crime spot and same will be forwarded to the Police Side so that the police will reach the crime spot as early as Possible.

Precondition: The user must login into the App and must Report about the Crime where the crime is actually Committed.

Assumption: Crime can be Committed at any Point in any area ir-respective of a Police station Nearby so our App will be able to report about Crime easily.

Test Steps:

1. Login with the registered UserName and password
2. App will view about reporting Crime option
3. There is an option of capturing images.

4. The GPS will be automatically fetch the location.
5. User must switch on the GPS option in order to fetch location.

Expected Result: The location must be fetched wherever the App is Used.

Actual Result:

The Location are actually fetched of the Nearest Police station.

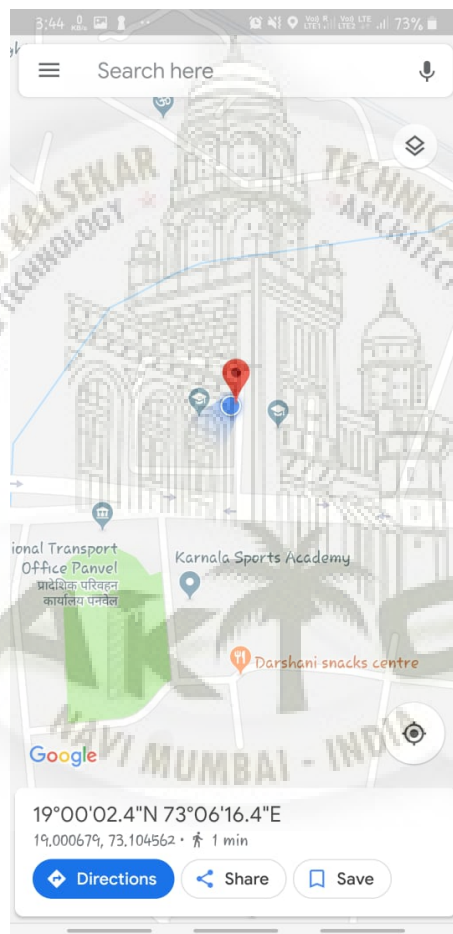
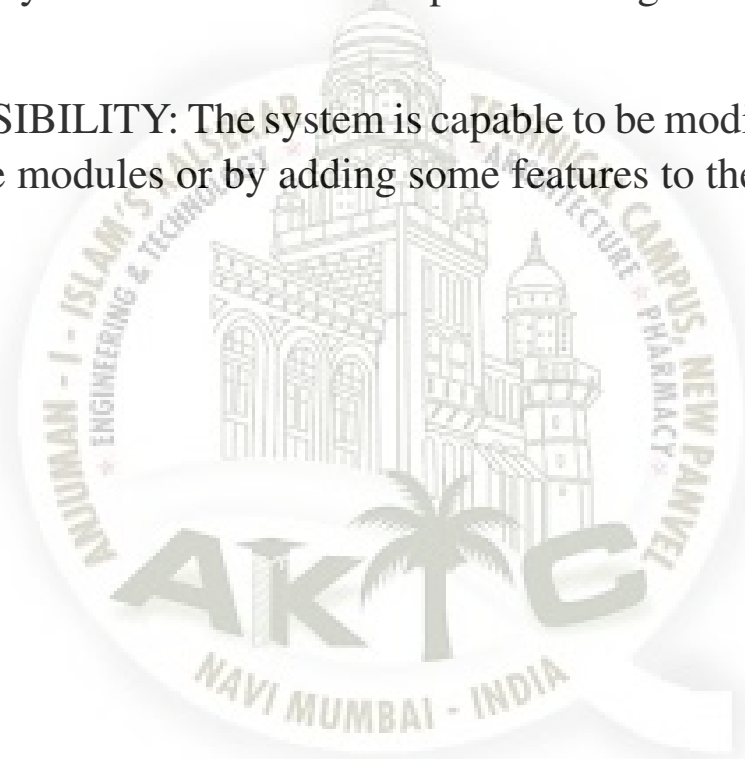


Figure 7.3: Fetching Location

7.2.1 Software Quality Attributes

1. **AVAILABILITY:** The system should not be down, whenever the user use the system the specific data should be available to the user.
2. **CORRECTNESS:** As per the user search the correct data should be shown to the user like at time for searching the near by place the system should show only the places around the user.
3. **MAINTAINABILITY:** The administrators of the system will maintain the system with effective updates though on air update if needed.
4. **EXTENSIBILITY:** The system is capable to be modified by changing some modules or by adding some features to the existing system



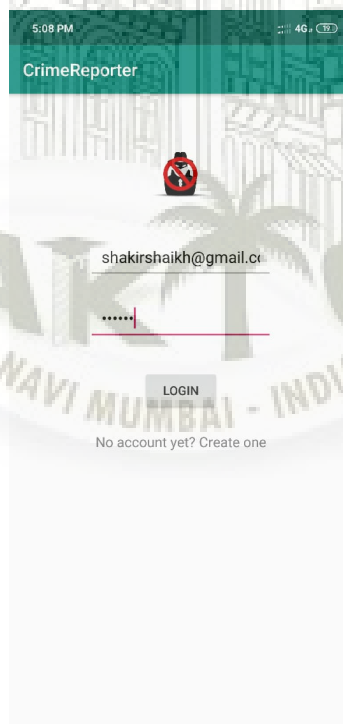
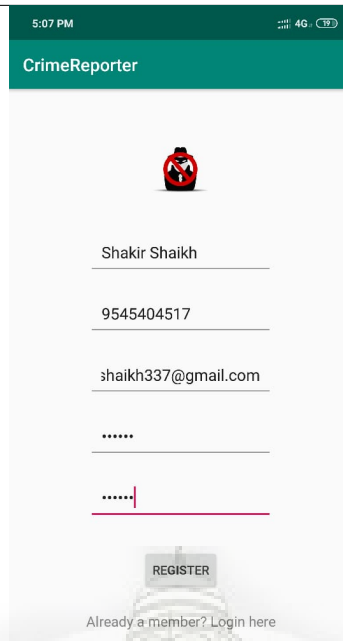


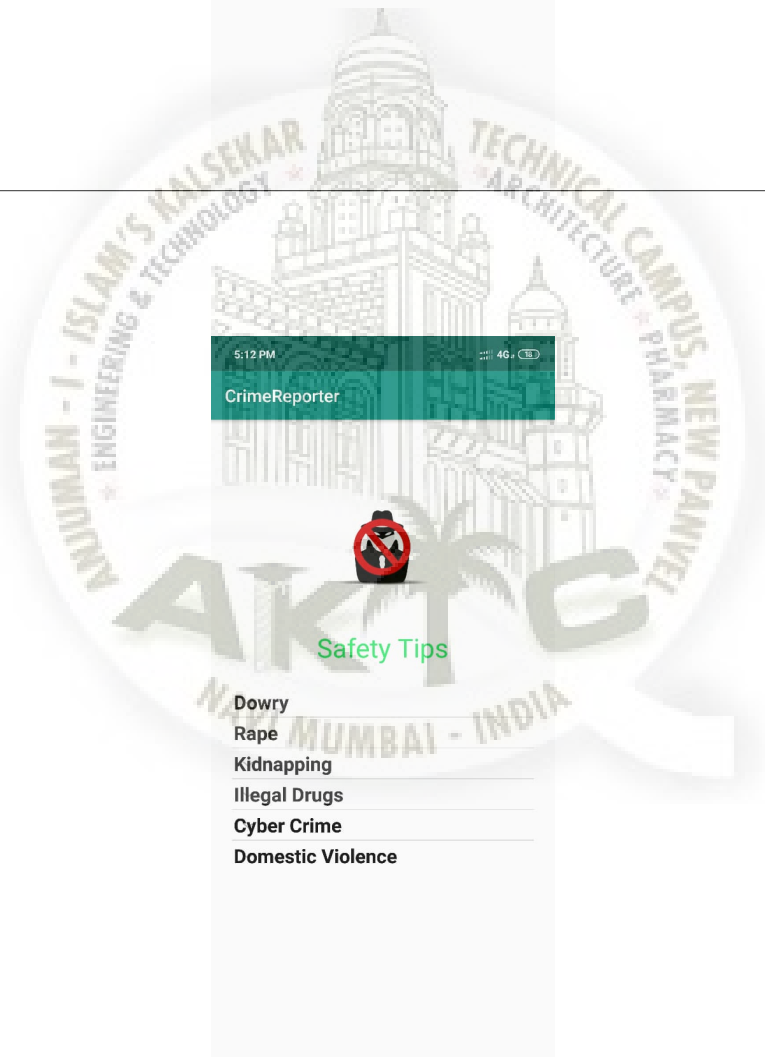
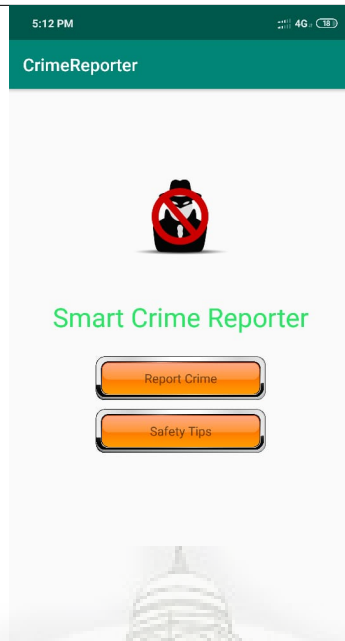
Chapter 8

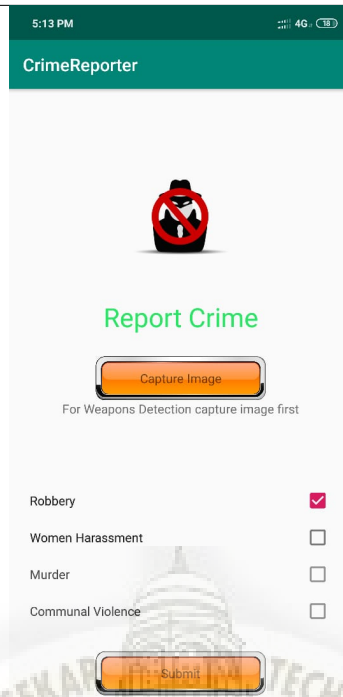
Screenshots of Project

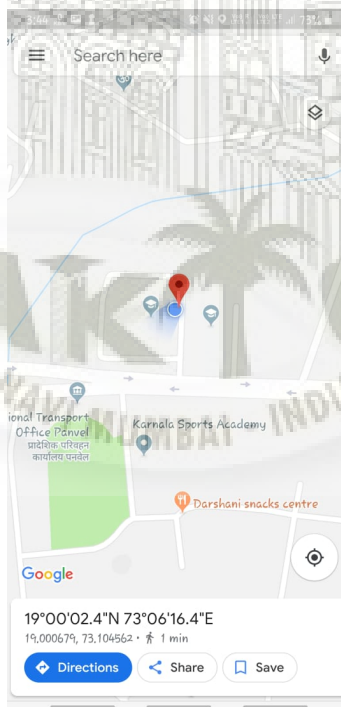
8.1 Smart Crime Reporter











Chapter 9

Conclusion and Future Scope

9.1 Conclusion

The Proposed System will be able to Report Crimes in more Efficient and quick manner it will Help those people who are afraid of reporting crimes because they are too busy in their own Work. User just have to login into App and Report the Crimes. This will help Police more efficiently to catch criminals also with the Help of Google maps the location is also traced through our system and same will be Forwarded to Police Panel.

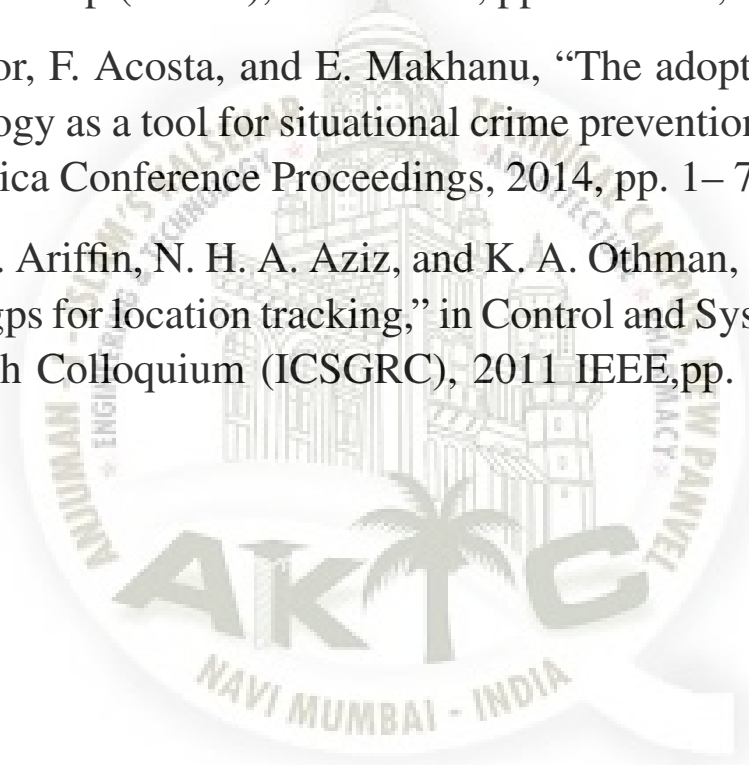
9.2 Future Scope

WRITE HERE.

- As of now we are trying to generate reporting call through input as spoken keyword or capture image.
- Our future scope is to identify fake calls with machine learning and implement fake call management.
- Also, we will add gesture which will trigger the app as soon we draw the figure For: Eg will draw S on Screen to open our app with gesture.
- Our one of the most important future scope is to manage a list of attended and unattended crimes by police department so further investigations could be carried on by higher authorities for negligence of unattended cases.

References

- [1] . H. Michaely, X. Zhang, G. Simko, C. Parada, and P. Aleksic, “Keyword spotting for google assistant using contextual speech recognition,” in Automatic Speech Recognition and Understanding Workshop (ASRU), 2017 IEEE, pp. 272–278, IEEE, 2017.
- [2] C. Oduor, F. Acosta, and E. Makhanu, “The adoption of mobile technology as a tool for situational crime prevention in kenya,” in IST-Africa Conference Proceedings, 2014, pp. 1– 7, IEEE, 2014.
- [3] A. A. B. Ariffin, N. H. A. Aziz, and K. A. Othman, “Implementation of gps for location tracking,” in Control and System Graduate Research Colloquium (ICSGRC), 2011 IEEE, pp. 77–81, IEEE, 2011.



Achievements

1. Project Competitions

- (a) ”*Smart Crime Reporter*”; Shaikh Ayub, Shaikh Shakir, Safa Maldar; Avalon 2019(National Level Technical Paper Presentation, March 2019)
(Venue: New Mumbai)



TERNA PUBLIC CHARITABLE TRUST'S
TERNA ENGINEERING COLLEGE
AN ISO 9001 : 2000 | NBA ACCREDITATION



CERTIFICATE OF PARTICIPATION

This is to certify that

Ayub Shaikh

of

A.I. Kalsekar Technical Campus

has participated in

Avalon 2019, A National Level
(Technical Paper Presentation / Project Competition)
conducted on 5th & 6th March, 2019
at Terna Engineering College, Nerul




Prof. D.M. Bavkar
Avalon co-ordinator


Dr. L.K. Ragha
Principal

Figure 9.1: Paper Presentation Certificate- Ayub Shaikh



Figure 9.2: Paper Presentation Certificate- Shaikh Shakir



Figure 9.3: Paper Presentation Certificate- Safa Maldar