RV COLLEGE OF ENGINEERING®, BENGALURU - 560059

(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

“Missing Persons Database”

PROJECT REPORT
SOFTWARE ENGINEERING LAB(18IS55)
V SEMESTER

2020-2021

Submitted by

Sriram NC	1RV18CS173
Utkarsh Choubey	1RV18CS180
Simhendra Urs.	1RV18CS167

Under the Guidance of
Prof. Prapulla S.B.,

Assistant Professor, Department of Computer Science and Engineering,

RV College of Engineering,

Bengaluru

Academic Year 2020- 2021

RV COLLEGE OF ENGINEERING®, BENGALURU - 560059

(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the **Mini-**Project work titled “**Missing Persons Database**” has been carried out by Sriram NC (1RV18CS173), Utkarsh Choubey (1RV18CS180) andSimhendra Urs (1RV18CS167) bonafide students of RV College of Engineering, Bengaluru, have submitted in partial fulfillment for the **Assessment of Course: SOFTWARE ENGINEERING(18IS55) Laboratory** during the year 2020-2021. It is certified that all corrections/suggestions indicated for the internal assessment have been incorporated in the report.

Faculty Incharge
Department of CSE,
RVCE., Bengaluru –59

Head of Department
Department of CSE,
RVCE, Bengaluru–59

RV COLLEGE OF ENGINEERING®, BENGALURU - 560059
(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We, **Sriram NC (1RV18CS173), Utkarsh Choubey (1RV18CS180), Simhendra Urs (1RV18CS167)** the students of 4th Semester B.E., Department of Computer Science and Engineering, R.V. College of Engineering, Bengaluru hereby declare that the mini-project titled **"Missing Persons Database"** has been carried out by us and submitted in partial fulfillment for the **Assessment of Course: SOFTWARE ENGINEERING(18IS55)** during the year 2020-2021.

Place: Bengaluru.

Sriram NC (1RV18CS173)
Utkarsh Choubey (1RV18CS180)
Simhendra Urs (1RV18CS167)

ACKNOWLEDGEMENT

Any achievement, be it scholastic or otherwise does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends. A number of personalities, in their own capacities have helped me in carrying out this project work. I would like to take this opportunity to thank them all.

I deeply express my sincere gratitude to my guide **Prapulla SB , Associate Professor Department of CSE, RVCE**, Bengaluru, for her able guidance, regular source of encouragement and assistance throughout this project.

I would like to thank **Dr.Ramakanth Kumar P**, Head of Department, Computer Science & Engineering, R.V.C.E, Bengaluru, for his valuable suggestions and expert advice.

First and foremost I would like to thank **Dr. Subramanya K N**, Principal, R.V.C.E, Bengaluru, for his moral support towards completing my project work.

I thank my Parents, and all the Faculty members of the Department of Computer Science & Engineering for their constant support and encouragement.

Last, but not the least, I would like to thank my peers and friends who provided me with valuable suggestions to improve my project.

Abstract

Most investigating agencies in India use offline records to track missing persons, which involves a lot of manpower and resources. This consumes precious time in which the victims may disappear forever. Missing persons database is a system which is used by detective agencies to find people who have been missing for a long time such as victims of kidnapping or murder. It usually records information about the victims prior to their disappearance due to various possible reasons. Sometimes this can be in the form of an FIR. There is a need for improvement in the current system which should be efficient and reliable which can aid the problem and this is what motivated us to do this project.

We propose an intelligent, online method of investigating people's disappearance in which most of the work is done automatically and the police only need to follow up using the results of our system, which contains a GUI, uses facial recognition, GPS location tagging and tracking, etc. to make an accurate prediction on the location of the victim. The Missing Persons Tracing System is a web application built on Django framework based on python while the frontend is built on HTML, CSS and JavaScript. This uses both RDBMS - MySQL as well as NoSQL - MongoDB. This system is basically a follow-on to the current system of tracking cases of kidnapping, disappearance, etc. It can be seen as an improvement from the current system as it also has room for automation. The main need to make this is to embrace newer technologies for database storages and how they can be seen as better alternatives than the usual databases. It also shows how automation can be incorporated into the system.

As a result of our self-study, we were able to implement the concepts taught to us, along with the materials we found on the web. We successfully created a full-fledged website, using which the user can order, and sell used and new products. We successfully implemented NoSQL integration and the frontend using Django. The whole project was completed after 8 weeks of hard work by the team members. Our gratitude to the faculty who helped us in this process with their valuable guidance and encouragement.

Table of Contents

Chapter 1

Introduction

Missing persons database is a system which is used by detective agencies to find people who have been missing for a long time such as victims of kidnapping or murder. It usually records information about the victims prior to their disappearance due to various possible reasons. Sometimes this can be in the form of an FIR.

1.1 Objective

The Missing Persons Tracing System is basically a follow-on to the current system of tracking cases of kidnapping, disappearance, etc. It can be seen as an improvement from the current system as it also has room for automation. The main need to make this is to embrace newer technologies for database storages and how they can be seen as better alternatives than the usual databases. It also shows how automation can be incorporated into the system.

1.2 Scope

Goal is to build an automated Missing Persons Tracing System. Objectives of the Missing Persons Tracing System are:

1. A system for police officers to digitally investigate missing persons reports and catch the suspects responsible.
2. A face recognition system which will be able to recognise victims via surveillance to track their location faster.

The benefits of this system are :

1. It will be very easy to understand the functionalities and operate upon them.
2. Investigation can be automated to a large extent and will reduce the use of manpower and resources and result in speedy justice.
3. The facial recognition system will mitigate the need for police officers to manually cross- reference pictures in an offline database.

1.3 Proposed System

We propose an intelligent, online method of investigating people’s disappearance in which most of the work is done automatically and the police only need to follow up using the results of our system, which contains a GUI, uses facial recognition, GPS location tagging and tracking, etc. to make an accurate prediction on the location of the victim.

1.4 Societal Concern and Innovative Component

Our concern is improvement of law and order to ensure that people who are missing for any reason can be found easily and the perpetrators can be caught and punished accordingly. We are using technologies like facial recognition to track the last known location of the victim as well as surveillance footage of the possible criminal involved. We can also use social media for NLP to analyse mysterious cases of disappearance.

Chapter 2

Software Requirement Specification

2.1 Introduction

2.1.1 Purpose

a. The purpose of the S.R.S is to present a detailed description of the software for the tracing of missing persons. It will explain the purpose and features of the software, the interfaces of the software, what the software will do and the constraints under which it must operate. This S.R.S is intended for users of the software and also potential developers. This S.R.S will also provide a reference to verify whether the developed software meets the intended objectives, at any phase of development

2.1.2 Document Conventions

This SRS is created based on the IEEE template for System Requirement Specification Documents.

2.1.3 Intended Audience and Reading Suggestions

- The Missing Persons Tracing System can be used by police officers, detective agencies, volunteers, etc to track down people who disappeared due to any unknown reason.

- PROGRAMMERS: 1.1 purpose,1.5 Product scope,2.2 Product functions, 2.4 Operating environment,2.5 Design and implementation constraints 2.6 User documentation,3.External Interface Requirements ,4 .System features, 5.Other non functional requirements,6.Other requirements

- ADMINISTRATOR(senior police officers at a police station): 1.Introduction,2.2 Product functions, 2.4 Operating environment,2.6 User documentation,3.External Interface Requirements ,4.System features,5.Other non functional requirements,6.Other requirements

2.1.4 Product Scope

Goal is to build an automated Missing Persons Tracing System. Objectives of the Missing Persons Tracing System are:

1. A system for police officers to digitally investigate missing persons reports and catch the suspects responsible.
2. A face recognition system which will be able to recognise victims via surveillance to track their location faster.

The benefits of this system are :

1. It will be very easy to understand the functionalities and operate upon them.
2. Investigation can be automated to a large extent and will reduce the use of manpower and resources and result in speedy justice.
3. The facial recognition system will mitigate the need for police officers to manually cross reference pictures in an offline database.

References

1. Technologies for Identifying Missing Children, Final Report, ANSER Analytic Services Inc. 186277
2. https://www.news18.com/news/india/ncrb-launches-portal-for-searching-missing-persons-check-police-record-of-vehicles-2478719.html
3. https://www.livemint.com/companies/news/staqu-uses-ai-to-help-police-find-missing-children-1556692683030.html

2.2 Overall Description

2.2.1 Product Perspective

The Missing Persons Tracing System is basically a follow-on to the current system of tracking cases of kidnapping, disappearance, etc. It can be seen as an improvement from the current system as it also has room for automation. The main need to make this is to embrace newer technologies for database storages and how they can be seen as better alternatives than the usual databases. It also shows how automation can be incorporated into the system.

2.2.2 Product Functions

Login Module: Login will be done through a secure login system wherein an authenticated police officer can login with their credentials after registration with important information such as their name, associated police station, etc.

Web Application: The Missing Persons Tracing System will be in the form of a Web application, where the police officer can update or access critical information about the case and the system would also predict location of the victim and suspect involved.

Reporting System: Users shall be able to file reports online about details of disappearance.

Case Status: User shall be able to view the status of the case and update details and also send messages to the assigned police officers.

2.2.3 User Classes and Characteristics

Such is the importance and simplicity of the subject of this software, that it is expected to be used by users of various classes, very frequently. Still, some classes of users that will look to use the functionality provided by this software more often than others are:

1. **Technical Staff** : These are the ones who will supervise the overall functionality of the system. They keep checks so that records are up to date and also the maintenance of the system
2. **Police** : They are the ones who attain the benefits of the system by taking its help for locating missing persons.
3. **Victims:** For filing online FIRs with details about the missing person.
4. **Administrators:** To read or modify any database records and to assign officers to cases.

2.2.4 Operating Environment

Environments on which the software will operate are :

- Client / Server System
- Operating System: Windows 10, Mac or Linux
- Database: SQL + MongoDB
- Platform: Django/Python
- Latest version of any browser i.e Chrome Chrome 86, Safari 14, Edge 86, Firefox 82 Internet Explorer 11

2.2.5 Design and Implementation Constraints

Following are be the constraints on the design and implementation of the proposed software:

1. If many people try to access the database there may be a lag in the application. This may be due to the inherent properties of the database.
2. The programs may be bundled in such a way that the cohesion is quite high . This may break the code further down in the future when the code is refactored.
3. The api used to recognize faces might not be 100% accurate but works for most of the time.
4. Implement the database at least using a centralized database management system. If the database server fails the whole application fails. But this scenario is quite rare as the database is handled by a third party organization who provides pretty good fault tolerance.
5. The api for face recognition may be degraded further down the line and soon become obsolete but this scenario is also quite rare.

2.2.6 User Documentation

The project can be found in the following Git repo: https://github.com/sriramcu/MissingPersonsTracing.

2.2.7 Assumptions and Dependencies

The software is to be developed using python and deployed as a web application on Django. For back-end storage and management, it requires MySQL, MongoDB . Programmers or testers, who are looking to examine the correctness of the software, should have the above-mentioned softwares installed on their system. This applies to Windows, Mac as well as Linux users.

2.3 External Interface Requirements

2.3.1 User Interfaces

- Front-end software: Django
- Back-end software: MySQL, MongoDB

2.3.2 Hardware Interfaces

- Windows, Mac, Linux.
- A browser which supports CGI, HTML & JavaScript.
- An Internet connection

2.3.3 Software Interfaces

Facial recognition will aid the software to identify the suspect and the victim. This will be implemented with the help of python modules. To add a new person into the system the person should take photos and add it to the MongoDB database and rest of the heavy lifting is done by the api for facial recognition. This api is chosen such that the accuracy is high and also doesn’t turn out to be an overhead on the main implementation itself i.e the database. Lastly, all the above-mentioned services will be realized and assembled in an application deployed on Django.

2.3.4 Communications Interfaces

The proposed software supports almost all kinds of browsers. We are using simple forms for the various stages of authentication. All the standard internet protocols, such as HTTPS, FTP, DNS etc. will be supported by the developed software.

2.4 System Features

This section demonstrates Missing Persons Tracing System’s most prominent features and explains how they can be used and the results they will give back to the user.

2.4.1 Accessing Criminal Records by police officers

2.4.1.1 Description and Priority

Priority : High

Usage : Frequent

Description : This is an important feature by which the police officer can find out information about past suspects to aid their investigations.

Risk : Security issues- We don’t want unauthorised users using such sensitive data.

2.4.1.2 Stimulus/Response Sequences

Criminal details must be displayed in a neat, formatted manner.

2.4.1.3 Functional Requirements

Any latest browser is required for this operation along with MySQL server and MongoDB for facial recognition via CCTV footage.

2.4.2 Accessing Victim Reports by police officers

2.4.2.1 Description and Priority

Priority : Highest

Usage : Frequent

Description : This is an important feature by which the police officer can find out information about missing persons and relative/witness info to aid their investigations.

Risk : Security issues- We don't want unauthorised users using such sensitive data.

2.4.2.2 Stimulus/Response Sequences

Victim details must be displayed in a neat, formatted manner.

2.4.2.3 Functional Requirements

Any latest browser is required for this operation along with MySQL server and MongoDB for facial recognition via CCTV footage.

2.4.3 Registration

2.4.3.1 Description and Priority

Priority : Medium

Usage : Moderate

Description : This is a feature to register police officers by making them enter their name, mobile number(verified by OTP), police station ID, etc.

Risk : Security issues- We don't want unauthorised users entering false information to gain login credentials.

2.4.3.2 Stimulus/Response Sequences

User data should be entered into a form and a success message should be displayed on entering into our secure database.

2.4.3.3 Functional Requirements

Any latest browser is required for this operation along with MySQL server.

2.4.4 Login

2.4.4.1 Description and Priority

Priority : Medium

Usage : Moderate

Description : This is a feature to login and check officer's authorization.

Risk : Security issues- We don't want unauthorised users entering false information to gain login credentials.

2.4.4.2 Stimulus/Response Sequences

User data should be entered into a form and a success message should be displayed on entering into our secure database.

2.4.4.3 Functional Requirements

Any latest browser is required for this operation along with MySQL server.

2.4.5 Filing cases/ checking case status

2.4.5.1 Description and Priority

Priority : High

Usage : High

Description : This is a feature to login and check officer's authorization.

Risk : Security issues- We don't want unauthorised victims filing cases and sending messages to police officers. At the same time it is impractical to require victims to register an account to lodge an FIR. Thus, we can store the source IP address for every message sent.

2.4.5.2 Stimulus/Response Sequences

User data should be entered into a form and a success message should be displayed on entering into our secure database.

2.4.5.3 Functional Requirements

Any latest browser is required for this operation along with MySQL server.

2.4.6 Facial recognition

2.4.6.1 Description and Priority

Priority : Low

Usage : Low

Description : This is a feature to match surveillance footage with known images in the database.

Risk : Low

2.4.6.2 Stimulus/Response Sequences

User data should be entered into a form and a success message should be displayed on entering into our secure database.

2.4.6.3 Functional Requirements

Any latest browser is required for this operation along with MySQL server.

2.5 Other Nonfunctional Requirements

2.5.1 Performance Requirements

Performance will depend upon the speed and quality of the internet connection as well. Any interface between a user and the Missing Persons Tracing System should have a maximum response time of 5 seconds. The response should be fast enough to avoid users' response collisions. The Missing Persons Tracing System should be available for use 24 hours per day, 365 days per year. The Missing Persons Tracing System should support 500 patrons and 1000 requests/min simultaneously

2.5.2 Safety and Security Requirements

It is essential that the system is not prone to hacking and that only authorized users have access to login credentials. All information is sensitive and it should not be possible to copy data to the clipboard, take screenshots of information or otherwise extract the information out of the system.

2.5.3 Software Quality Attributes

The software is scalable to bigger organizations. It will be able to handle more requests due to the selection of the database taken into consideration. This is the inherent quality of the database itself and needs no effort from the side of developers or customers to fix this issue.

2.5.4 Business Rules

Any application that stores confidential information and allows access to users, has some rules regarding who can access what. Our system will give the permission of uploading the data into the database, for the first time to the authorities of the organization that are employing the functionality being provided by the software.

2.6. Other Requirements

Appendix A: Glossary

Python: Python is an interpreted, high-level, general-purpose programming language

Web framework: A web framework (WF) or web application framework (WAF) is a software framework that is designed to support the development of web applications

MySQL: MySQL is an open-source relational database management system

MongoDB: MongoDB is a cross-platform document-oriented database program. Classified as a NoSQL database program

Django: Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design.

Chapter 3

Design

3.1 Overview

After reviewing the Use Case analysis, following are the basic classes and actions that emerge:
Classes: (Basic building blocks of Online Missing Persons Tracing System)

Sl no.	**Class**	**Principle Responsibility**
1.	Police Officer	Avail functionalities of the application to trace missing persons.
2.	Missing Person	Avail functionalities of the application to file an FIR online.
3.	Sighting	Provides information about the last seen location of the victim.
4.	Suspect	Criminals in the police records, providing easy to use information for investigation.

Note: Other subsidiary classes may get added to the list in course of implementation for the purpose of load balancing and modularity.

Actions:

Sl No.	Action
1.	Register a case with sightings.
2.	View or modify criminal or victim records.
3.	Search for criminals or victims using facial recognition.
4.	Update or view case status.

Note: There are other minor actions that do not play a major role in modeling.

3.2. System Structure

Here we describe the final structure. It should, however, be kept in mind that obtaining the final structure is an iterative exercise – an initial structure is refined as the design progresses.
In particular, the dynamic modeling has an impact on the structure.

3.2.1 Inheritance Structure

There is one evident case of an inheritance structure that is between the types of users. There are 2 identified types of users- the Victims and the Police Officer. Here victim refers to a relative of the person gone missing. The figure below shows the inheritance structure.

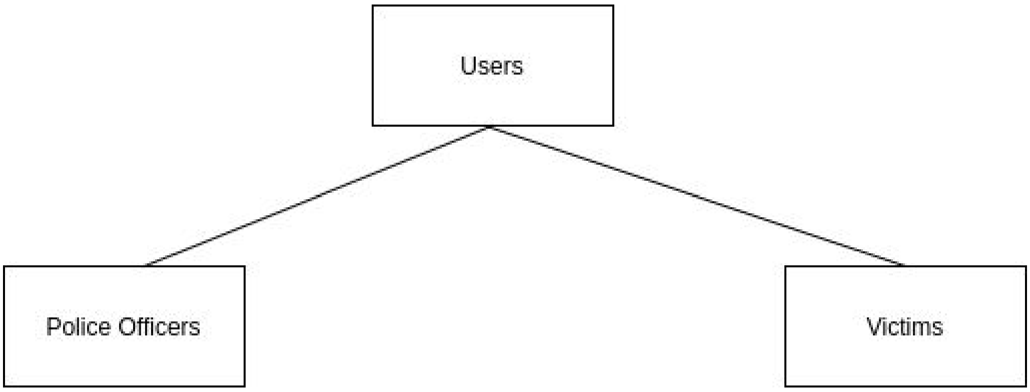


Fig 3.1 Inheritance Diagram

However, these inheritance structures are not necessary as the functionalities of all the users are different.

3.2.2 Entity Relationship Diagram

ER Diagram stands for Entity Relationship Diagram, also known as ERD is a diagram that displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. It is a high-level conceptual data model diagram. It is based on the notion of real-world entities and the relationships between them. ER modeling helps you to analyze data requirements systematically to produce a well-designed database. So, it is considered a best practice to complete ER modeling before implementing your database.
ER diagrams are created based on three basic concepts: entities, attributes and relationships.

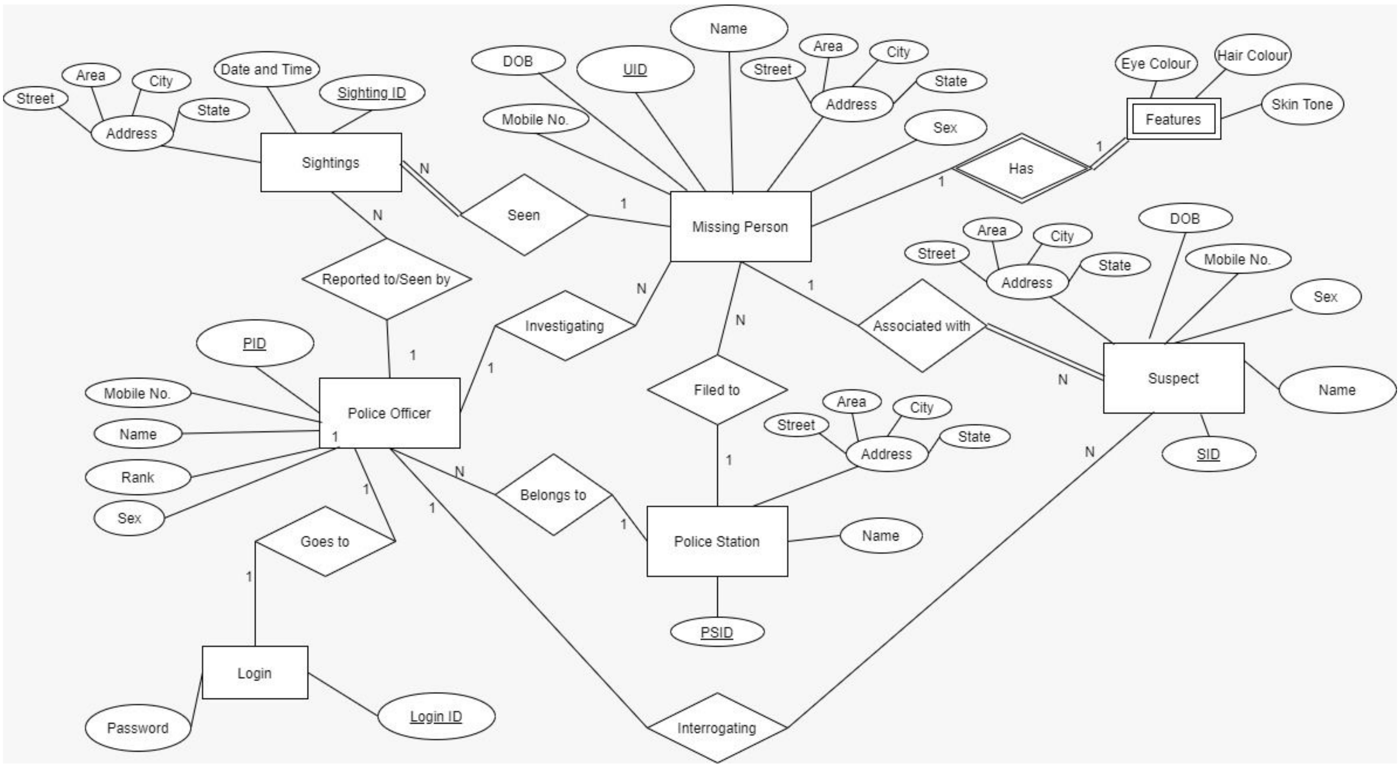


Fig 3.2 ER Diagram

Entities: Police Officer, Sightings, Missing Person, Suspect, Login, Features, Police Station.
Relationships: Reported to, Goes to, belongs to, investigating, filed to, seen, associated with, has.
In our project, we have a victim related to the missing person(PK:UID) who was seen many times in different places and associated with multiple suspects(PK:SID) investigated by the assigned police officer(PK:PID).
Every police officer belongs to one police station (PK:PSID) and has a login ID for our Django web application.

3.2.3 Complete class diagram

Finally, after considering all the major actions, the complete association and aggregation structure is arrived at.

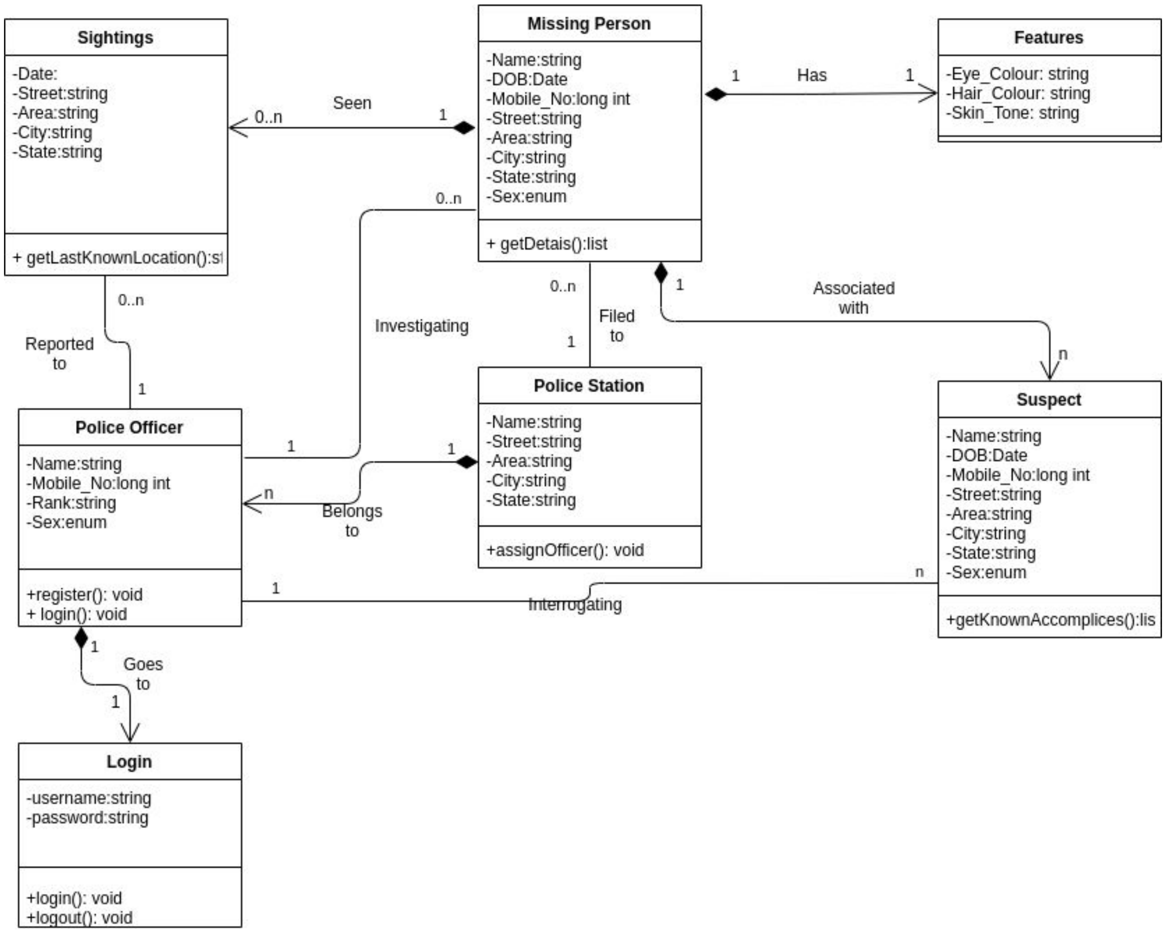


Figure 3.3 Class Diagram

3.2.4 Structure Chart

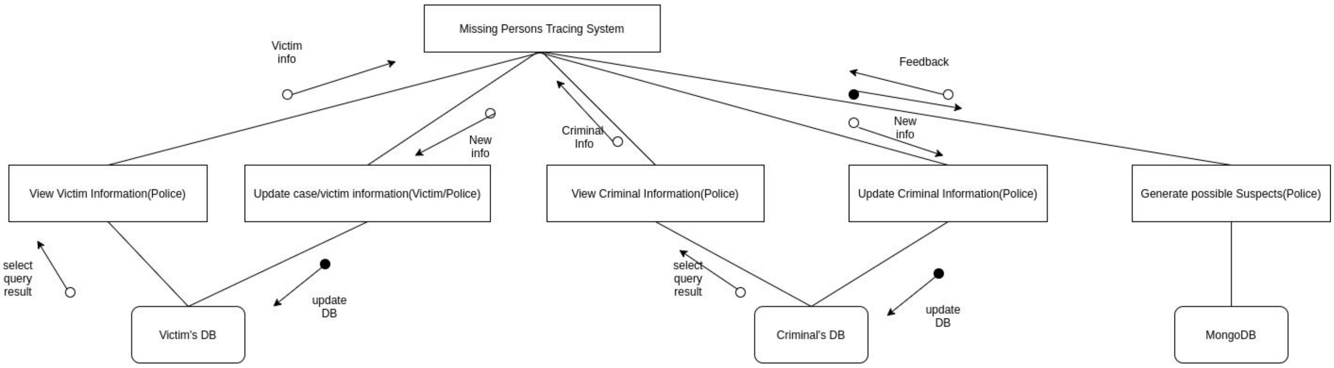

Missing Persons Tracing System
Victim
info
Feedback
New
info
Criminal
Info
New
info
View Victim Information(Police)
Update case/victim information(Victim/Police)
View Criminal Information(Police)
Update Criminal Information(Police)
Generate possible Suspects(Police)
select
query
result
update
DB
select
query
result
update
DB
Victim's DB
Criminal's DB
MongoDB

Figure 3.4 Structure Chart

3.3. System behavior

3.3.1 Sequence Diagram

The dynamic behavior of the system is modeled by figuring out the interactions between the classes involved in each principal action. We are showing the final diagrams here. It should be remembered that these models have an impact in refining and enhancing the class diagrams– we are not discussing these aspects here.

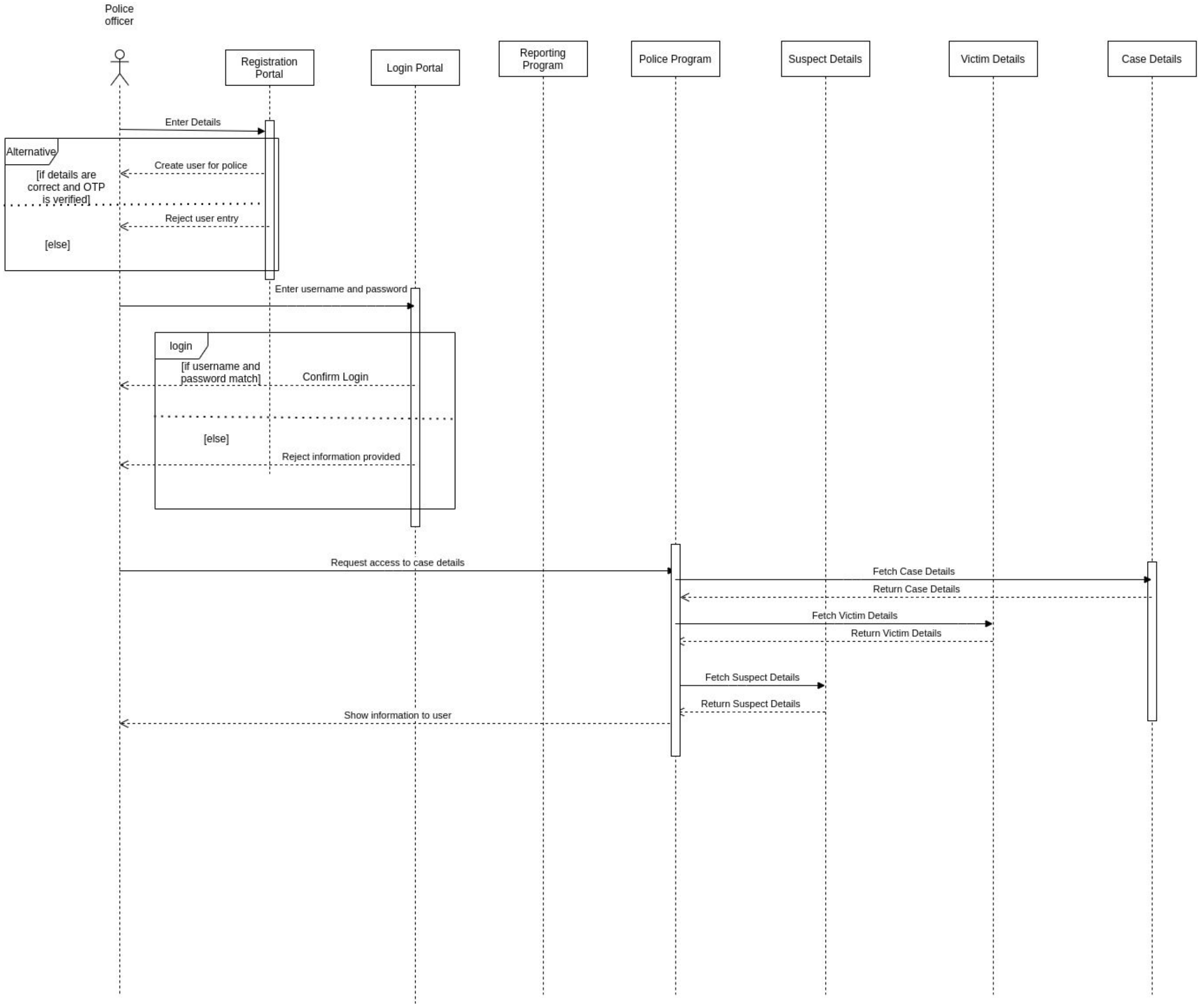


Figure 3.5 Sequence Diagram For Police Officer

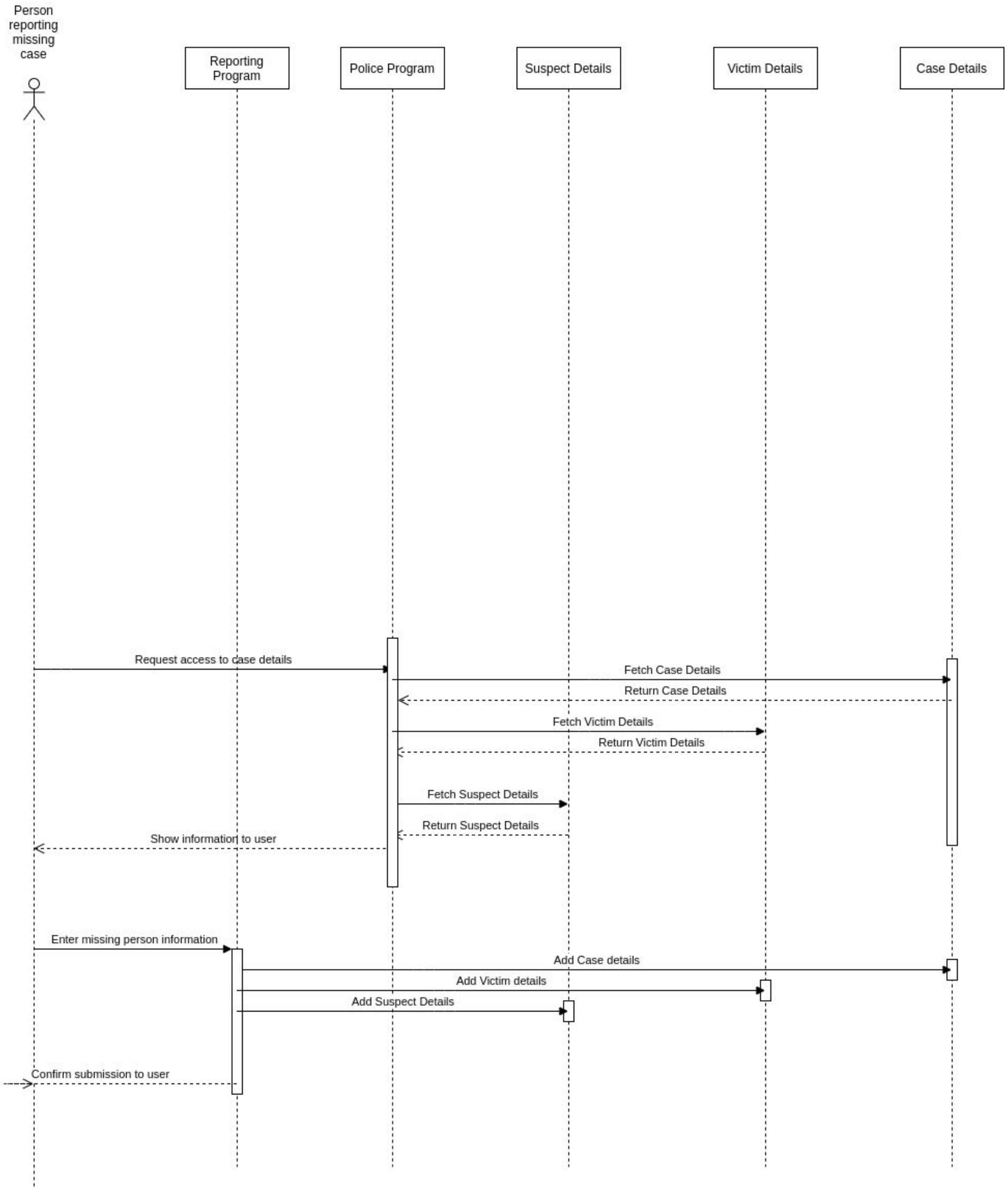


Figure 3.6 Sequence Diagram for Victim

3.3.2 Data flow diagram

Such a diagram represents the dynamic flow of data in the system, between various entities, processes and data stores.

Level 0 DFD

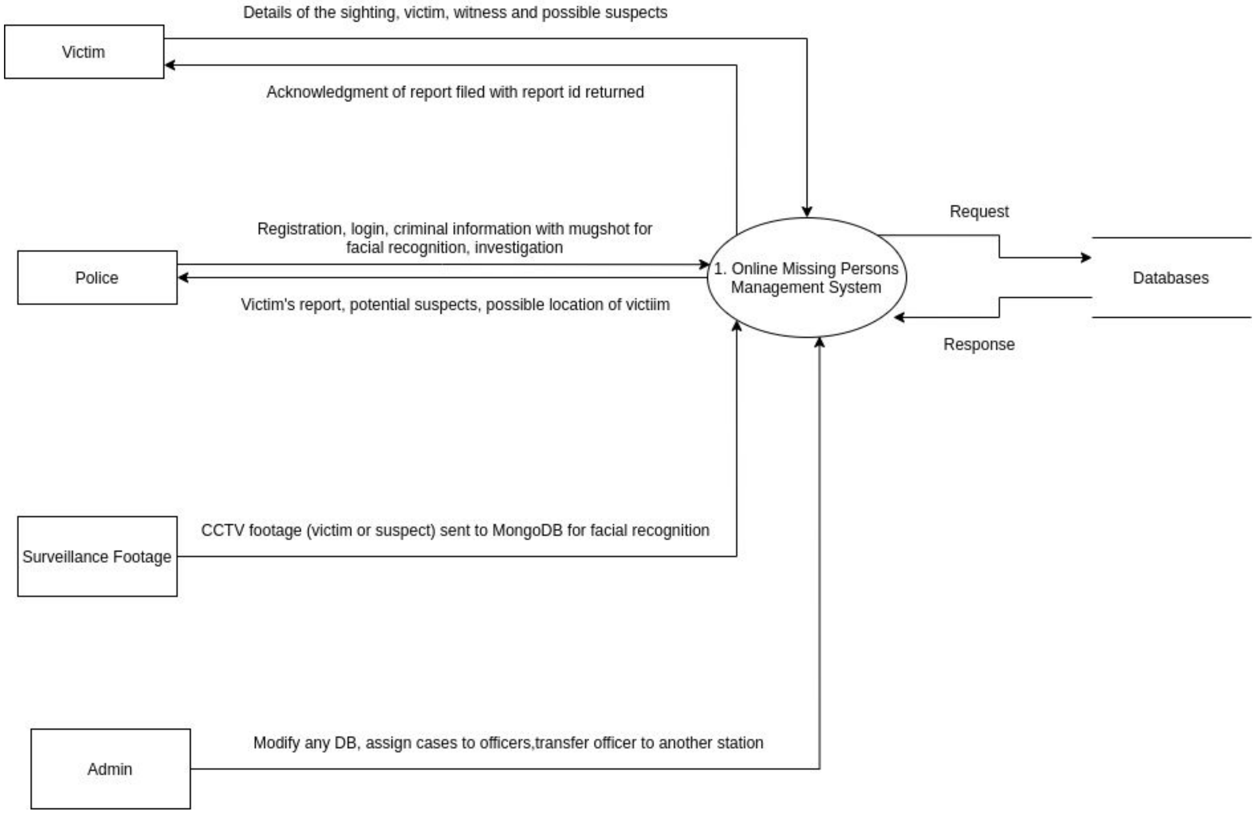

Details of the sighting, victim, witness and possible suspects
Victim
Acknowledgment of report filed with report id returned
Request
Registration, login, criminal information with mugshot for facial recognition, investigation
Police
1. Online Missing Persons Management System
Databases
Victim's report, potential suspects, possible location of victiim
Response
CCTV footage (victim or suspect) sent to MongoDB for facial recognition
Surveillance Footage
Modify any DB, assign cases to officers,transfer officer to another station
Admin

Figure 3.7 Level 0 DFD

Our system has 4 types of users- victim, police, surveillance footage and admin who use this online system and each feature involves either reading from or writing to our database.

Level 1 DFD

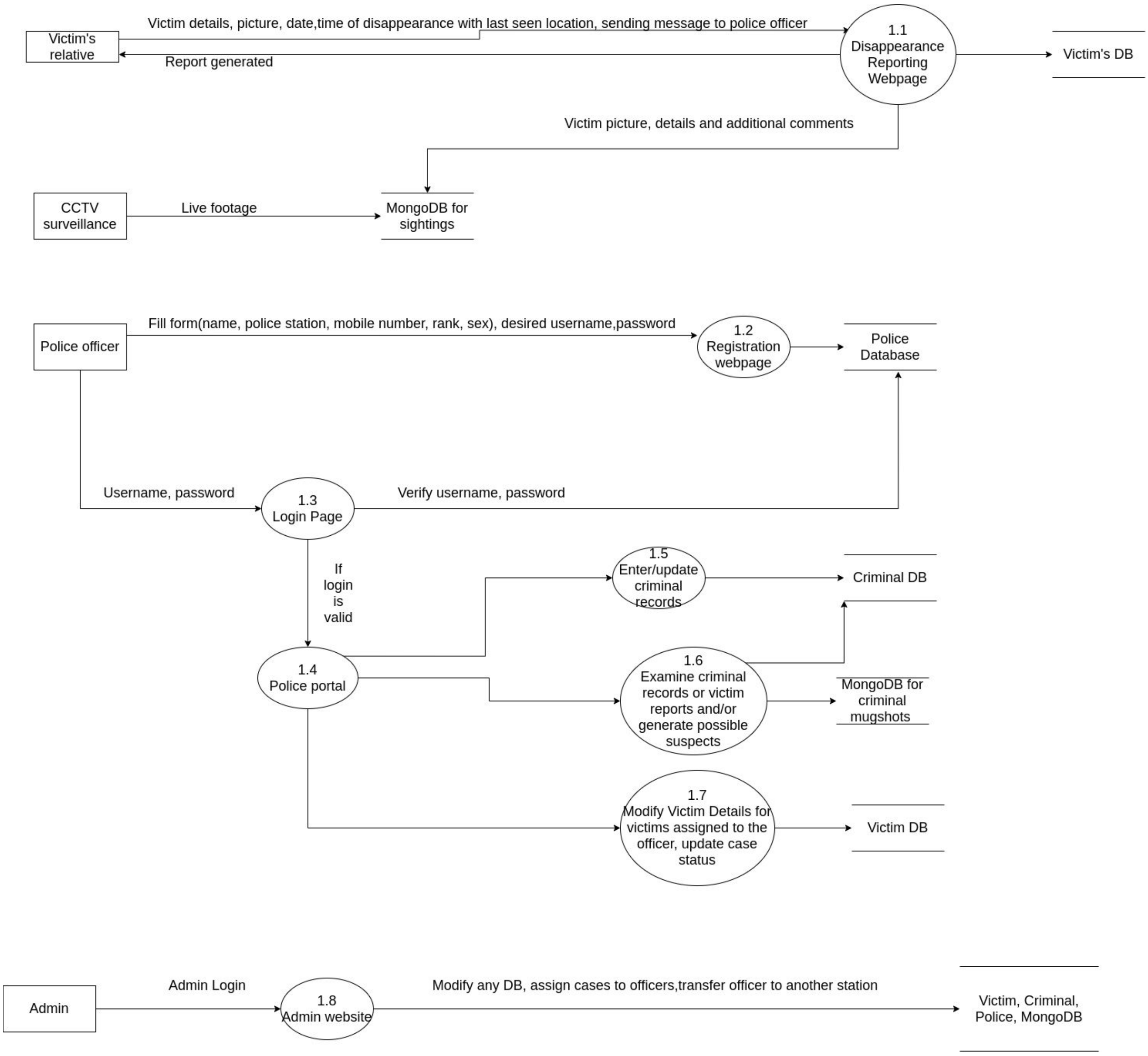

Victim's relative
Victim details, picture, date,time of disappearance with last seen location, sending message to police officer
Report generated
1.1 Disappearance Reporting Webpage
Victim's DB
Victim picture, details and additional comments
CCTV surveillance
Live footage
MongoDB for sightings
Police officer
Fill form(name, police station, mobile number, rank, sex), desired username,password
1.2 Registration webpage
Police Database
Username, password
1.3 Login Page
Verify username, password
If login is valid
1.4 Police portal
1.5 Enter/update criminal records
Criminal DB
1.6 Examine criminal records or victim reports and/or generate possible suspects
MongoDB for criminal mugshots
1.7 Modify Victim Details for victims assigned to the officer, update case status
Victim DB
Admin
Admin Login
1.8 Admin website
Modify any DB, assign cases to officers,transfer officer to another station
Victim, Criminal, Police, MongoDB

Figure 3.8 Level 1 DFD

The admin has the highest privilege. He can read, modify, add or delete any record in any database and can also assign cases to the right officer or transfer them where necessary and also maintain the security of the DB.

The police officer (who is not an admin) has to register with his details, verify the OTP and then login through the police portal's login page to read or modify criminal records or those of victims' assigned to him by the admin. The officer can also upload images to MongoDB or perform facial recognition with a given piece of CCTV surveillance footage. Finally, the officer can update the case status or close the case. All operations are logged on the server and can be seen by the Admin.

The victim's relative or the police can register a case or FIR of a missing person online and can also view the case status or send messages to the police officer assigned. The message is logged in the database with the victim's IP address to check authenticity.

3.4 Use Case Diagram

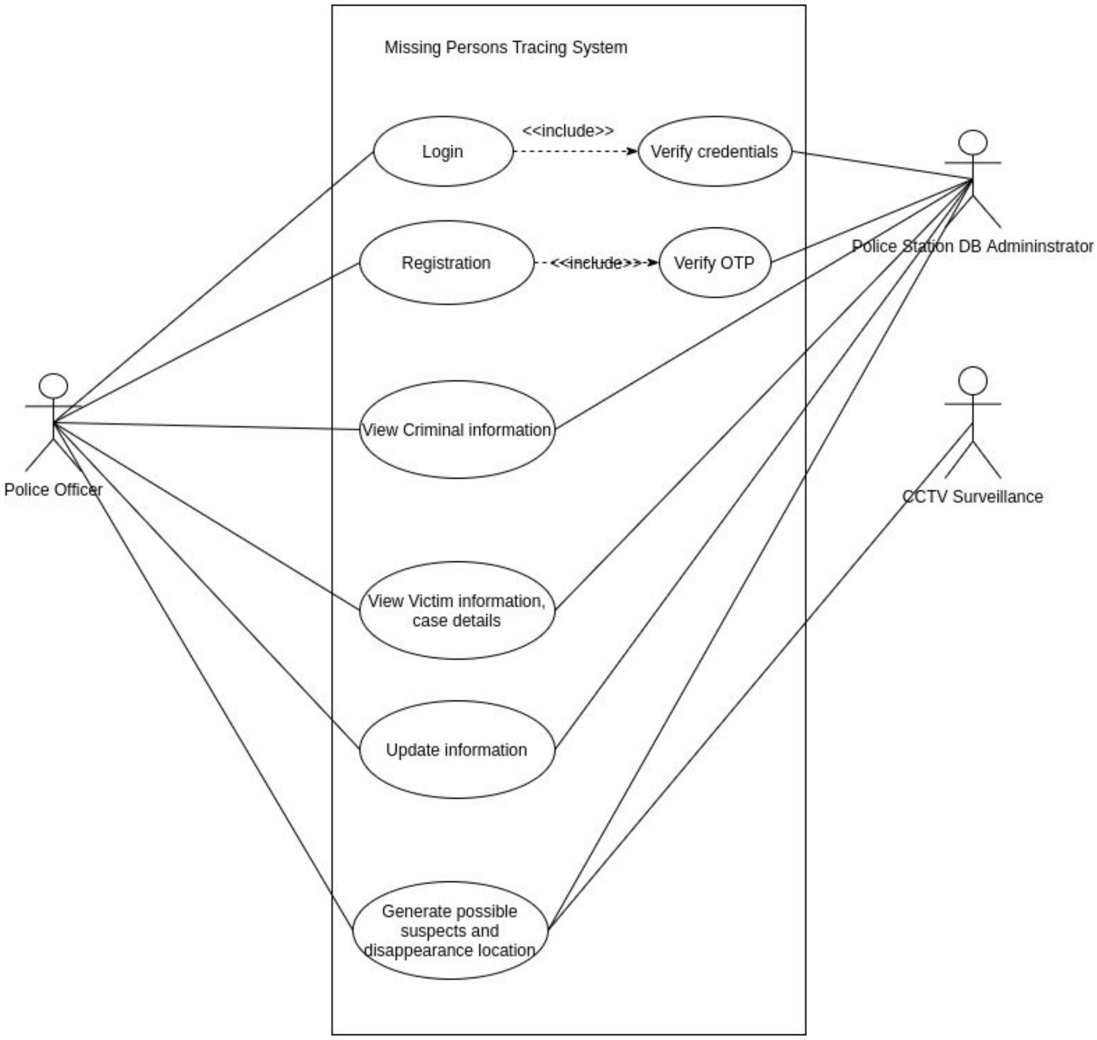

Missing Persons Tracing System
Login
<<include>>
Verify credentials
Registration
<<include>>
Verify OTP
Police Station DB Admininstrator
View Criminal information
Police Officer
CCTV Surveillance
View Victim information, case details
Update information
Generate possible suspects and disappearance location

Figure 3.9 Use Case Diagram for Police Officer

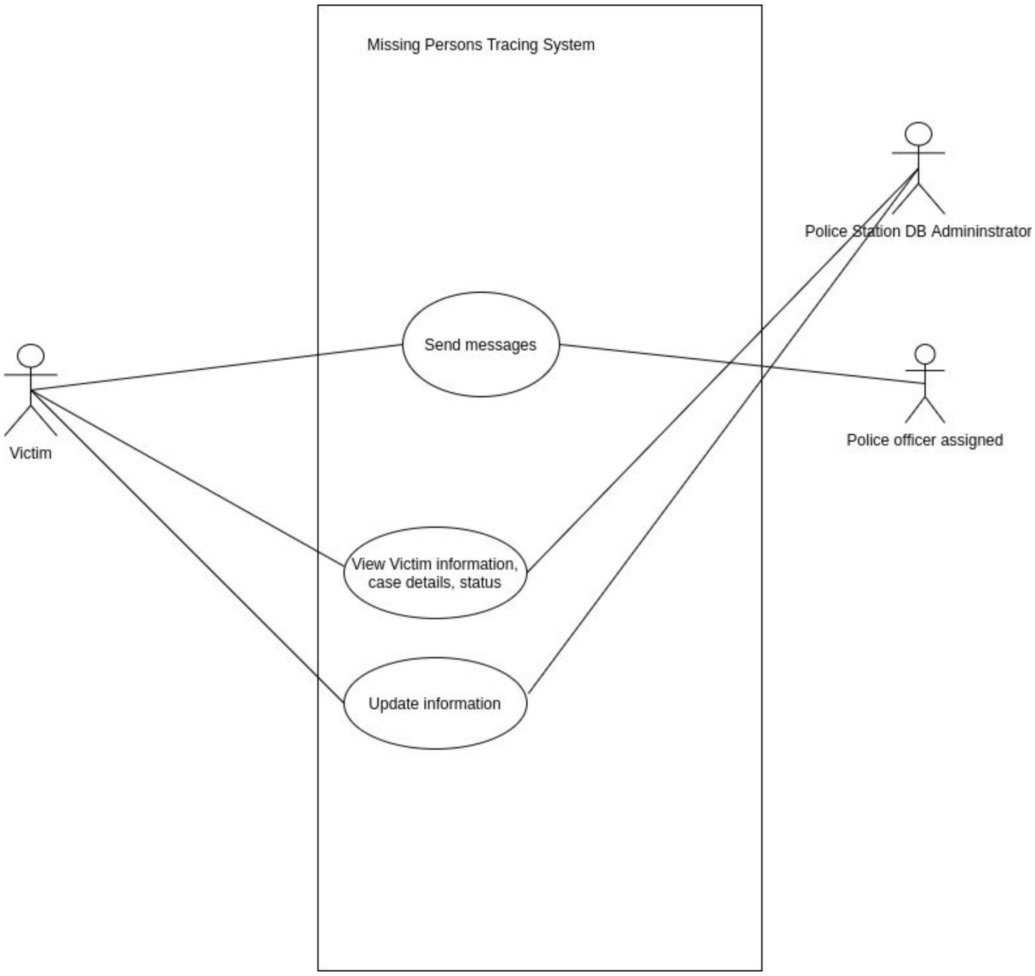

Missing Persons Tracing System
Police Station DB Admininstrator
Send messages
Victim
Police officer assigned
View Victim information, case details, status
Update information

Figure 3.10 Use Case Diagram for Victim

Chapter 4

Implementation Plan

4.1 Management Overview

4.1.1 Description of Implementation

The team applied an Agile SDLC Methodology, which is a combination of iterative and incremental process models, with a focus on process adaptability and improvement by rapid delivery of working software applications. An incremental approach was adopted, and the working software was built one feature at a time. The implementation process obeyed the following Agile Manifesto principles:

● Individuals and Interactions - In Agile development, self-organization and motivation are important, as are interactions for pair programming.

● Working Software - Demo working software was developed to analyze and understand the requirements better instead of just depending on documentations.

● Customer Collaboration - As the requirements cannot be gathered completely in the beginning of the project due to various factors, continuous self-validation and team interaction is very important to get proper product requirements.

● Responding to change - Agile development is focused on quick responses to change and continuous development/improvement.

4.1.2 Major Tasks

This section provides a description of the major system implementation tasks:

- Setup database configuration and connectivity
- Login module: This allows existing users to login and new users to create an account. New users are prompted to register.

- Case Registration Module: This allows either the victim or the police officer to register a case of a Missing Person with the last known sightings.
- Facial Recognition Module: To identify victims and suspects from a surveillance image.
- Case Status Module: To allow victims to view the status of their FIR and send messages to the assigned police officers.

- Run required tests and security implementation.

4.1.3 Security and Privacy

This section provides an overview of the system security and requirements that must be followed during implementation:

- Multiple validations are in place when the user is creating his account and logging in, so it prompts the user to enter a strong password, and the validity of his details is also verified.
- Password storage is done in a hashed method, so that sensitive information is not directly stored in the database.
- IP address is traced while registering cases and sending messages to ensure authenticity.

4.2 Implementation Support

This section describes any support hardware, software, facilities, and materials required for the implementation, along with the documentation, necessary personnel and training requirements, outstanding issues, and implementation impacts to the current environment.

4.2.1 Requirement Specification

4.2.1.1 Hardware

This section provides a list of support equipment: This hardware may include computers, servers, peripheral equipment, simulators, emulators, other non-computer equipment as well as any network and data communication requirements.

- Intel i5-4590 processor equivalent or greater
- 8GB+ RAM

4.2.1.2 Software

This section provides a list of non-hardware components (software, databases, and compilers, operating systems, utilities, etc.) required to support the implementation.

- Operating system: Linux
- MySQL database
- MongoDB
- HTML, CSS

4.2.1.3 Database Requirements

- MongoDB
- MySQL

4.2.1.4 Data Requirements

- Details of Users registered.
- Details of criminals and victims.
- Details of comments and messages.
- Images for facial recognition.

4.2.1.5 Functional Requirements

A functional requirement defines the functionality of a system or one of its subsystems. It also depends upon the type of software, expected users and the type of system where the software is used. The functional requirements of the proposed work are as follows:

1. Account creation, login and authorization

The police officer should be able to create an account by specifying a username, password, email id and mobile number. If any of the details provided are incorrect, or if the username already exists, or the password is too weak, or if there is a password mismatch, the user must be prompted with a suitable error message, and must be prompted to enter the correct details.

2. Case Registration: The victim or a police officer on their behalf should be able to easily file a case with or without an account with all relevant details, including place and time of last seen locations.

3. Case Status: The victim must be allowed to enter their unique details with a secure key sent to their mobile or email (like an OTP) to access the case status and to send messages to the assigned police officer.

4. Facial Recognition: The police officer must be allowed to either input images of known criminals or regular people from DL records to populate a database or to input surveillance footage to determine the most likely suspect and victim.

4.2.1.6 Non-Functional Requirements

The non-functional requirements of this project are:

1. Performance Requirements

The product shall be based on web and has to be run from a web server.The product shall take initial load time depending on internet connection strength, and the quality of the video being streamed.

The performance shall, to a large extent, depend upon hardware components, internet connection and RAM of the user. The system must be interactive and the delays involved must be minimal. The delays in case of opening windows forms, of popping error messages and saving the settings or sessions must be minimized. Even operations like opening databases, sorting questions and evaluation must not have significant delays.

2. Safety and SecurityRequirements

The username and password of every police officer must be stored in a database. However, the password must be stored in a hashed form to ensure security. Other information such as the email id and mobile number of the user must also be kept confidential. Also, when performing an operation, it must be ensured that the user's internet connection is secure.

3. Software Quality Attributes

The software must be reliable and available. While it is not a safety critical application, it must still be available throughout the day. Both experienced and inexperienced computer users must be able to use the application easily, and its user interface must be simple enough for anyone to understand.

4. Business Rules

Any application that stores confidential information of the users will have some rules regarding who can access what. Only authorized users (the administrators of the application) must be able to access the database. Most of the processes, however, must be automated.

4.2.2 Documentation

Following documentation is required to be delivered.

- Software requirements specification
- Use case Diagram
- ER Model
- Domain class Diagram
- Sequence diagram
- Data flow diagram
- Test coverage
- Test suite
- Design document

4.2.3 Societal Concern

In the current pandemic situation around the world, it is advised to avoid offline operations or to transition online. It is also more convenient in light of the growing number of parameters and data to be stored and analysed. Our concern is improvement of law and order to ensure that people who are missing for any reason can be found easily and the perpetrators can be caught and punished accordingly.

4.3 Implementation Requirements by Site

4.3.1 Site Implementation Details

4.3.1.1 Implementation details:

- A NoSQL database (MongoDB) has also been made use of for this project. It is being used to store the images for facial recognition. The advantage is that the images can be stored and accessed anywhere on the internet and not just on a central server.

● Python 3 has been used to interface to MongoDB, using the pymongo API with Atlas. We also used GridFS to store images in chunks for fast access.

● The system has a friendly UI for the users to interact easily with the frontend of the website easily and to interact with the webpage.

4.3.1.2 Risks and Contingencies

Following risks are identified in the project:

● Breach of confidential data from the database (Except password, which is protected by a one way hashing mechanism), SQL injection.

● Server overload due to high traffic.

● Server breakdown on DoS attack.

4.3.1.3 Implementation Verification and Validation

Following functionality must be present for validation:

Authentication

● Registration system for users

● Authentication system for users.

4.3.2 Acceptance Criteria

For transitioning the system into production, many of the requirements have to be filled which will lead to the smooth functioning of the system. Some of the criteria what we have identified in our project are described below:

- Server must be able to handle multiple requests within short periods of time.
- Security provided to the customers should be of very high quality and resist against Malware.
- The website should be active continuously, with very little down time, to cater to the customers' needs.
- The system should support a large number of users of different types at the same time.

Chapter 5

Testing

5.1 Introduction

The process or method of finding errors in a software application or program so that the application functions according to the end user's requirement is called software testing. Testing is evaluation of the software against requirements gathered from users and system specifications. Testing is conducted at the phase level in software development life cycle or at module level in program code.

A Test Plan is a detailed document that describes the test strategy, objectives, schedule, estimation, deliverables, and resources required to perform testing for a software product. Test Plan helps us determine the effort needed to validate the quality of the application under test. The test plan serves as a blueprint to conduct software testing activities as a defined process. Test Plan guides our thinking. It is like a rule book, which needs to be followed.

5.1.1 Purpose of the Test Plan Document:

This test plan keeps track of possible tests that will be run on the system after coding. Plan describes the testing approach and overall framework that will drive the testing of "Missing Persons Tracing System". Following objectives are achieved through this document.

- Test Strategy: It describes the rules on which the test will be based on, including the givens of the project description of the process to set up a valid test (e.g.: entry or exit criteria, creation of test cases, specific tasks to perform, scheduling, data strategy).
- Execution Strategy: It describes how the test will be performed and process to identify and report defects, and to fix and implement fixes.
- Test Management: It is the process to handle the logistics of the test and all the events that come up during execution (e.g.: communications, escalation procedures, risk and mitigation.

5.2 Test Suites

5.2.1 Police Authentication

Test Scenario ID	**Authentication 1**	**Test Case ID**	**Auth-1A**
Test case description	**Testing Authentication feature**	**Test Priority**	**High**
Pre-Requisites	**None**	**Post-Requisites**	**None**

Sl no	Action	Inputs	Expected output	Actual output	Test browser	Test result
1	Select Registration Hyperlink		Displays pop-up to enter details	Displays pop- up to enter details	Chrome	Pass
2	Enter username	sriramcu	Accepting string	Accepting string	Chrome	Pass
3	Enter Full Name	Sriram N C	String name accepted	String name accepted	Chrome	Pass
4	Enter Email-ID	csriram12345@gmail.com	Alphanumeric string with a '@'	Alphanumeric string with a '@'	Chrome	Pass
5	Enter Password	rvce_cse	Successful password with validation	Successful password with validation	Chrome	Pass
6	Click "Signup"		User registered successfully and redirected to the dashboard.	User registered successful and redirected to the dashboard.	Chrome	Pass
6	Select Login Hyperlink		Displays Login requireme nt s	Displays Login requiremen ts	Chrome	Pass
7	Enter username	sriramcu	Accepting alphanumeri c input	Accepting alphanumeric input	Chrome	Pass
8	Enter Password	rvce_cse	Accepting alphanumeri c password	Accepting alphanumeric password	Chrome	Pass
9	Sign-in		Checks database and logs in	Checks database and logs in	Chrome	Pass

10	**Wrong log-in credentials**		**Displays the issue**	**Displays the issue**	**Chrome**	**Pass**

5.2.2 Case Registration

Test Scenario ID	Case Details	Test Case ID	Application-1A
Test case description	Testing application input	Test Priority	High
Pre-Requisites	None	Post-Requisites	None

Sl. No	Action	Inputs	Expected output	Actual output	Test Browser	Test Result
1.	Select number of sightings	Input number from dropdown menu	Right number of sightings forms show up	Right number of sightings forms show up	Chrome	Pass
2	Enter full name	Utkarsh Choubey	String name input	String name input	Chrome	Pass
3	Enter email ID	utkarshchoubey15@gmail.com	Alphanumeric string with a '@'	Alphanumeric string with a '@'	Chrome	Pass
4	Enter first sighting details	<Place, date, time of sighting>	Input accepted	Input accepted	Chrome	Pass
5	Select Age	20	Numeric value accepted	Numeric value accepted	Chrome	Pass
6	Gender input	Male	Option button selected	Option button selected	Chrome	Pass
7	House No.	11	Numeric value accepted	Numeric value accepted	Chrome	Pass
8	City name	Bangalore	String input accepted	String input accepted	Chrome	Pass

9	**State name**	**Karnataka**	**String input accepted**	**String input accepted**	**Chrome**	**Pass**
10	**Submit Button**	**Submit Button Pressed**	**Data input into database**	**Data input into database**	**Chrome**	**Pass**

5.2.3 Check case status

Test Scenario ID	Status	Test Case ID	Status-1A
Test case description	Testing case status and message sending	Test Priority	High
Pre-Requisites	None	Post-Requisites	None

Sl. No	Action	Inputs	Expected output	Actual output	Test Browser	Test Result
1	Authenticate Victim	Select victim name and mobile number from dropdown menu- Utkarsh, 99999999999	Ask for secure key	Ask for secure key	Chrome	Pass
2	Display case status	Enter correct secure key, 0000	Case status displayed, can send message	Case status displayed, can send message	Chrome	Pass
3	Send message	Enter message, “Hello World”	Message stored in DB with victim’s IP	Message stored in DB with victim’s IP	Chrome	Pass

5.2.4 Facial Recognition

Test Scenario ID	Facial Recognition	Test Case ID	Facial-1A
Test case description	Testing facial recognition	Test Priority	Moderate
Pre-Requisites	Logged in Police Officer or Admin	Post-Requisites	None

Sl no	Action	Inputs	Expected output	Actual output	Test Browse r	Test Result
1	Upload image to DB	Image	Image successfully uploaded	Image successfully uploaded	Chrome	Pass
2	Upload surveillance image to check for matches	Surveillance Image	Matching person from DB, high confidence score	Matching person from DB, high confidence score	Chrome	Pass
3	Upload surveillance image to check for matches	Blurry surveillance Image	Matching person from DB, high confidence score	Fetched wrong person from DB, low confidence score	Chrome	Failed (due to blurry surveillance image)

5.2.5 Admin Authentication

Test Scenario ID	Admin Authentication 1	Test Case ID	AdminAuth-1A
Test case description	Testing Authentication feature	Test Priority	High
Pre-Requisites	None	Post-Requisites	None

Sl no	Action	Inputs	Expected output	Actual output	Test browser	Test result
1	Select Login hyperlink		Displays pop-up to enter details	Displays pop- up to enter details	Chrome	Pass
2	Enter Username	Admin	Accepting text string	Accepting text string	Chrome	Pass
3	Enter Password	Admin@12 3	Successful password with validation	Successful password with validation	Chrome	Pass
4	Login button committed		Display message about successful	Display message about successful	Chrome	Pass

5.2.6 Application Scrutinization

Test Scenario ID	Admin Scrutinization 1	Test Case ID	AdminScrutiny-1A
Test case description	Testing Scrutinization	Test Priority	High

Pre-Requisites	**None**	**Post-Requisites**	**None**

Sl. No	Action	Inputs	Expected output	Actual output	Test browser	Test result
1	Select table to modify	Reassign police officer	Modification successful	Modification successful	Chrome	Pass
2	Select table to modify	Delete police station	Police station and constituent police officers deleted	Police station and constituent police officers deleted	Chrome	Pass

Chapter 6

Conclusion and Future Enhancement

As a result of our self-study, we were able to implement the concepts taught to us, along with the materials we found on the web. We successfully created a full-fledged website, using which the police officers and victims can properly interact with the application to locate Missing Persons and bring the criminals to justice.

We successfully implemented backend web application on Django Framework - (Python), NoSQL (MongoDB) and MySQL databases and the frontend using HTML/CSS/JavaScript. The whole project was completed after 8 weeks of hard work by the team members. Our gratitude to the faculty who helped us in this process with their valuable guidance and encouragement.

There is a room for a lot of enhancement in this project. The web application UI/UX can be made even better by using standard libraries and the facial recognition feature could be made faster with the help of the cloud. Moreover, the bugs can be removed and the application can be made even better and complete with a better backend which is scalable, reliable and sustainable.

References

1. Rahimy Rahim, *Nearly 4000 Malaysia children went missing in 2014 and 2015*, March 2016, [online] Available: http://www.thestar.com.my/news/nation/2016/03/24/nearly-4k-missing-children-in-last-two-years/.

2. E. D. Kaplan, Understanding GPS: Principles and Applications, Artech House Publishers, February 1996, ISBN 0890067937.

3. Python Django Tutorial - https://youtu.be/JT80XhYJdBw

4. MongoDB Tutorial - https://youtu.be/-56x56UppqQ

5. https://www.stratoscale.com/blog/dbaas/hybrid-databases-combining-relational-nosql/

6. https://www.w3schools.com/whatis/

7. https://www.tutorialspoint.com/django/index.htm

8. https://lefred.be/content/top-10-reasons-for-nosql-with-mysql/

9. https://www.nfstc.org/programs/namus/index.html

10. https://trackthemissingchild.gov.in

11. https://www.javatpoint.com/python-mysql-database-connection

12. https://pymongo.readthedocs.io/en/stable/tutorial.html

13. https://www.news18.com/news/india/ncrb-launches-portal-for-searching-missing-persons-check-police-record-of-vehicles-2478719.html

14. https://beginnersbook.com/2015/05/normalization-in-dbms/

15. https://www.livemint.com/companies/news/staqu-uses-ai-to-help-police-find-missing-children-1556692683030.html

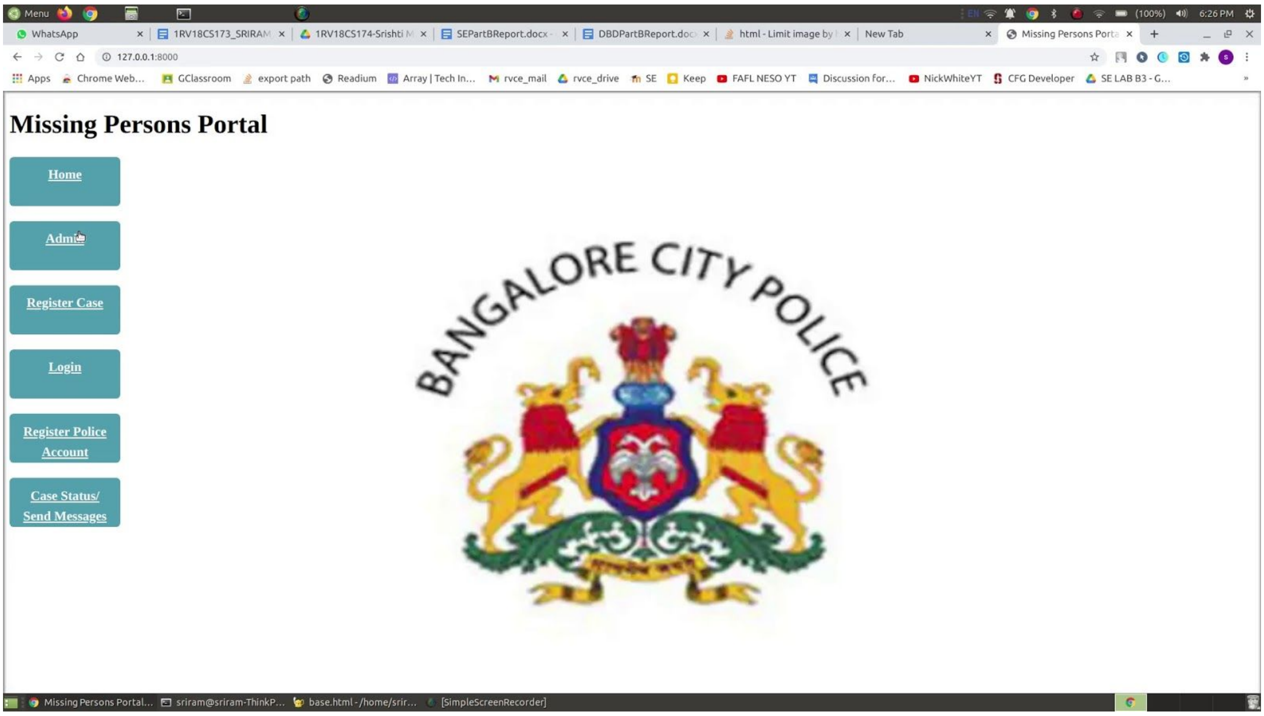
Appendix

Source Code

Github Link - https://github.com/sriramcu/MissingPersonsTracing

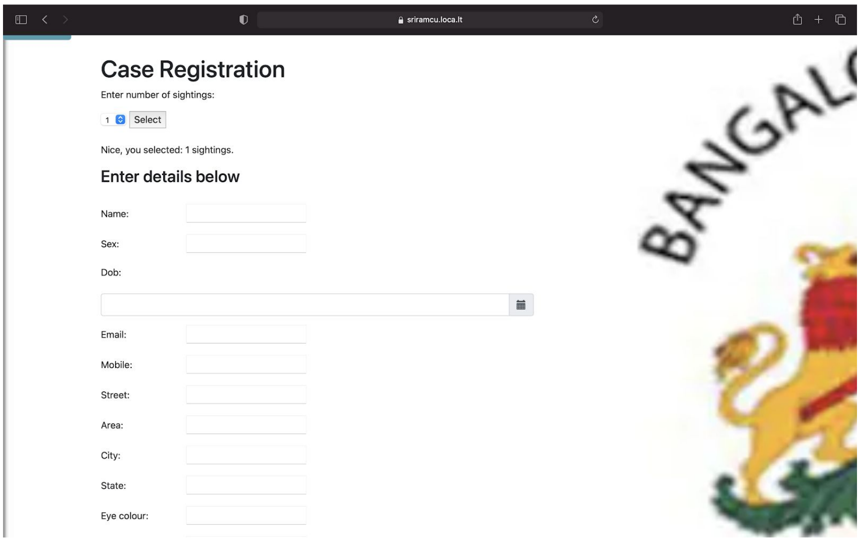
In the above link, we have put the snapshots and README for all features, including the ones that have been developed since the completion of this report.

Snapshots



Missing Persons Portal

Fig B.1 Portal Home Page



Case Registration

Enter number of sightings:

1 Select

Nice, you selected: 1 sightings.

Enter details below

Name:

Sex:

Dob:

Email:

Mobile:

Street:

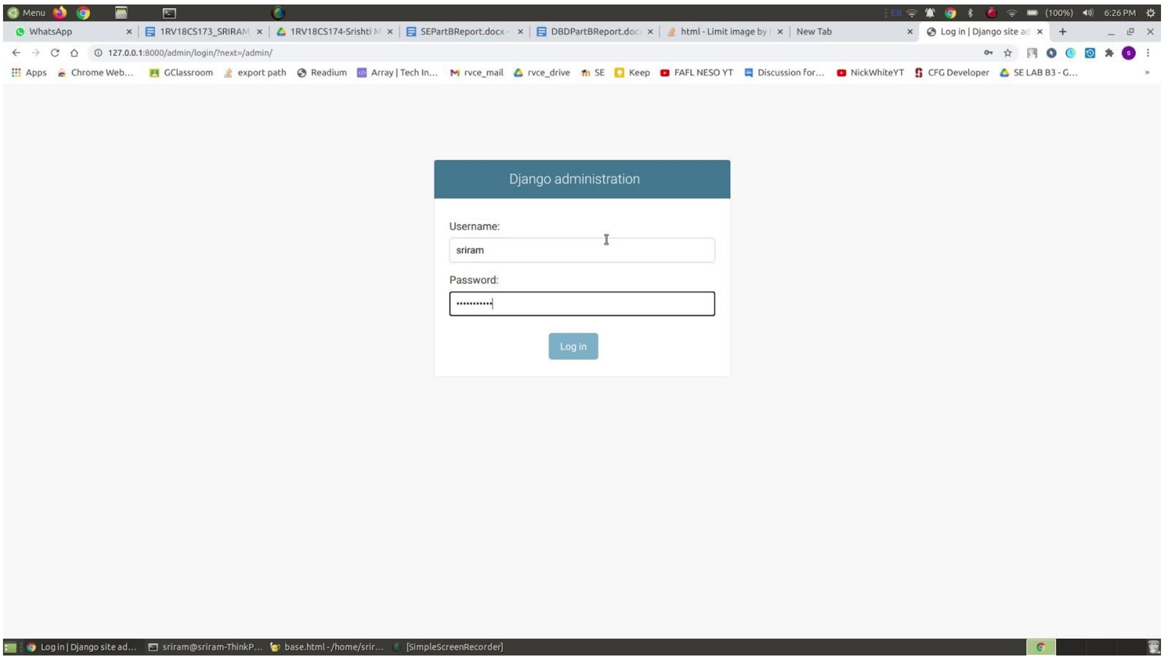
Area:

City:

State:

Eye colour:

Fig B.2 Victim Case Registration



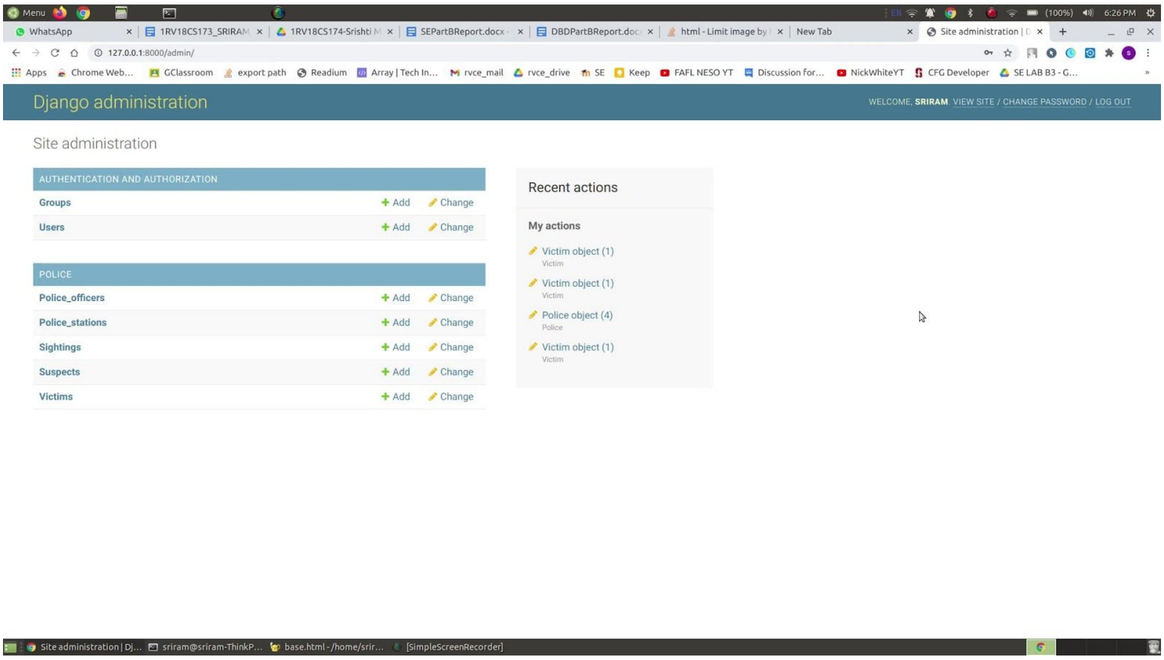
Django administration

Username: sriram

Password: •••••••••••

Log in

Fig B.3 Admin Login Page



127.0.0.1:8000/admin/

Django administration

WELCOME, SRIRAM. VIEW SITE / CHANGE PASSWORD / LOG OUT

Site administration

AUTHENTICATION AND AUTHORIZATION		
Groups	+ Add	Change
Users	+ Add	Change

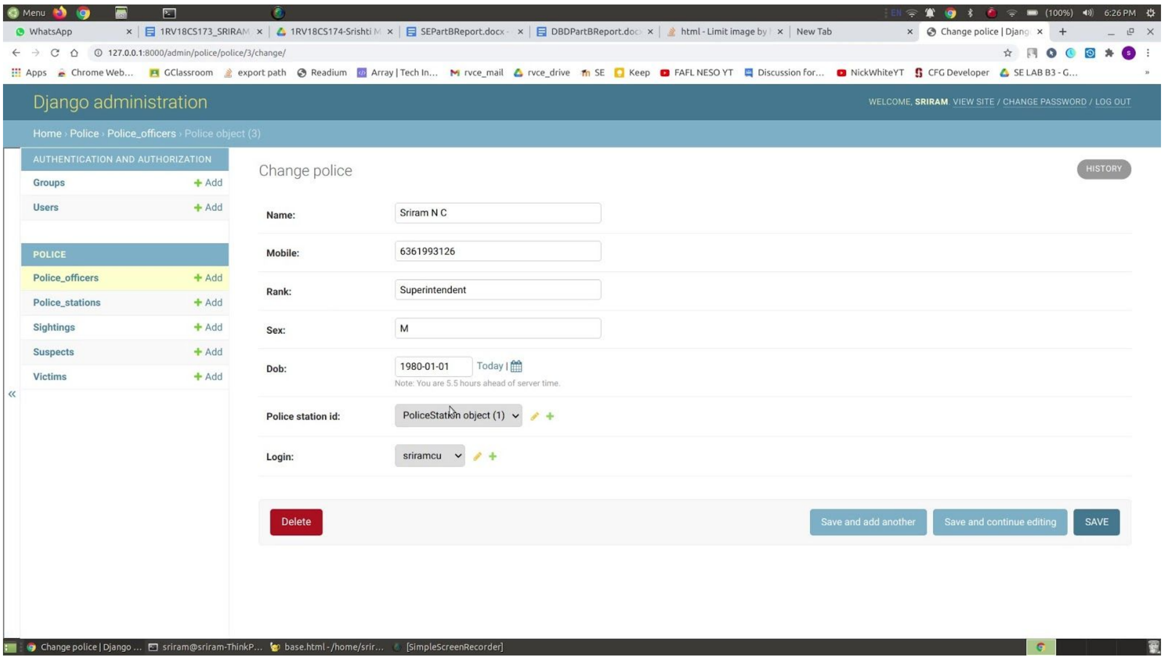
POLICE		
Police_officers	+ Add	Change
Police_stations	+ Add	Change
Sightings	+ Add	Change
Suspects	+ Add	Change
Victims	+ Add	Change

Recent actions

My actions

- Victim object (1)
  Victim
- Victim object (1)
  Victim
- Police object (4)
  Police
- Victim object (1)
  Victim

Fig B.4 Admin home page



127.0.0.1:8000/admin/police/police/3/change/

Django administration

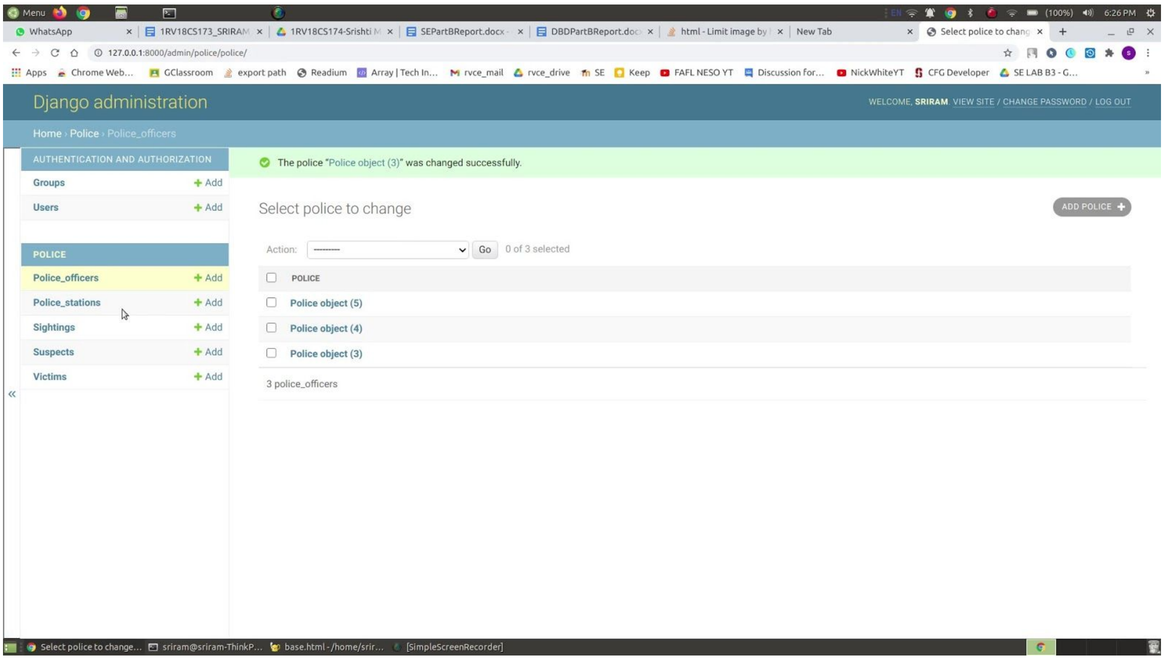
WELCOME, SRIRAM. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home › Police › Police_officers › Police object (3)

Change police

Name:	Sriram N C
Mobile:	6361993126
Rank:	Superintendent
Sex:	M
Dob:	1980-01-01 Today | Note: You are 5.5 hours ahead of server time.
Police station id:	PoliceStation object (1)
Login:	sriramcu

Fig B.5 Editing police officer’s station from 1 to 2.



127.0.0.1:8000/admin/police/police/

The police "Police object (3)" was changed successfully.

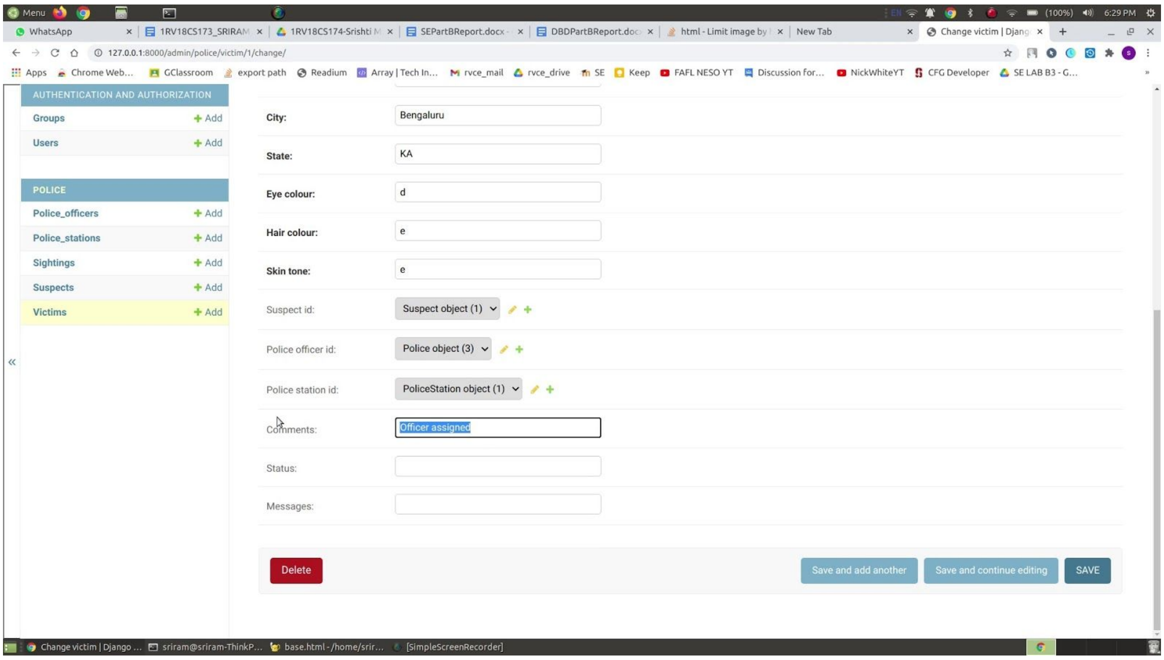
Select police to change

0 of 3 selected

□	POLICE
□	Police object (5)
□	Police object (4)
□	Police object (3)

3 police_officers

Fig B.6 Success Message after editing police station



127.0.0.1:8000/admin/police/victim/1/change/

City: Bengaluru

State: KA

Eye colour: d

Hair colour: e

Skin tone: e

Suspect id: Suspect object (1)

Police officer id: Police object (3)

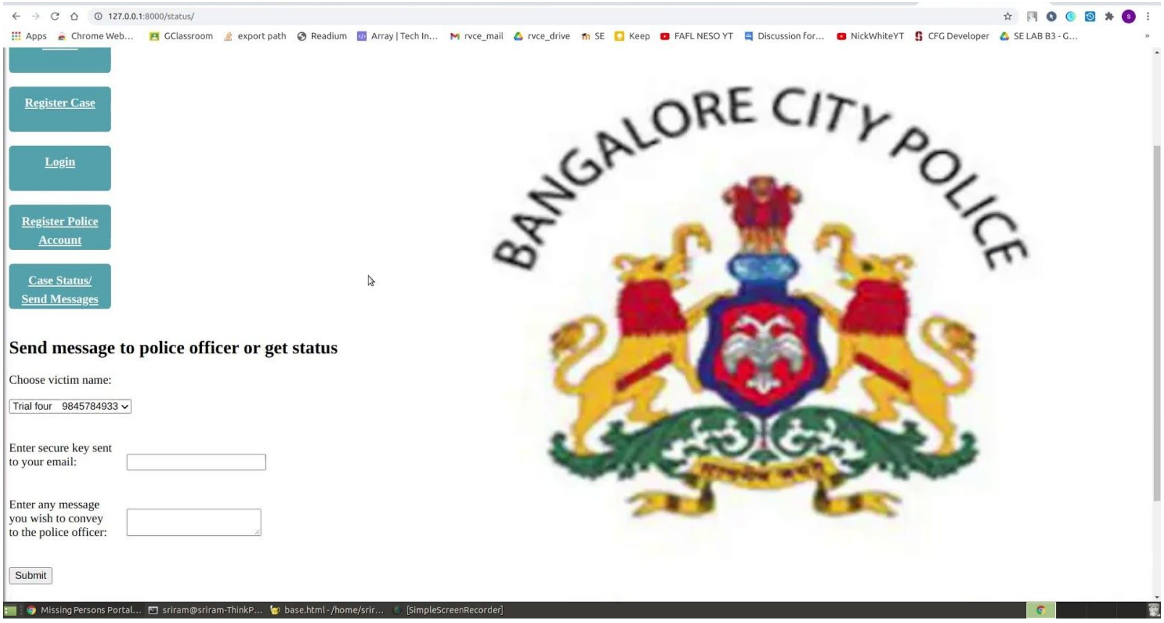
Police station id: PoliceStation object (1)

Comments: Officer assigned

Status:

Messages:

Fig B.7 Editing Victim comments and reassigning the police officer.



Send message to police officer or get status

Choose victim name:

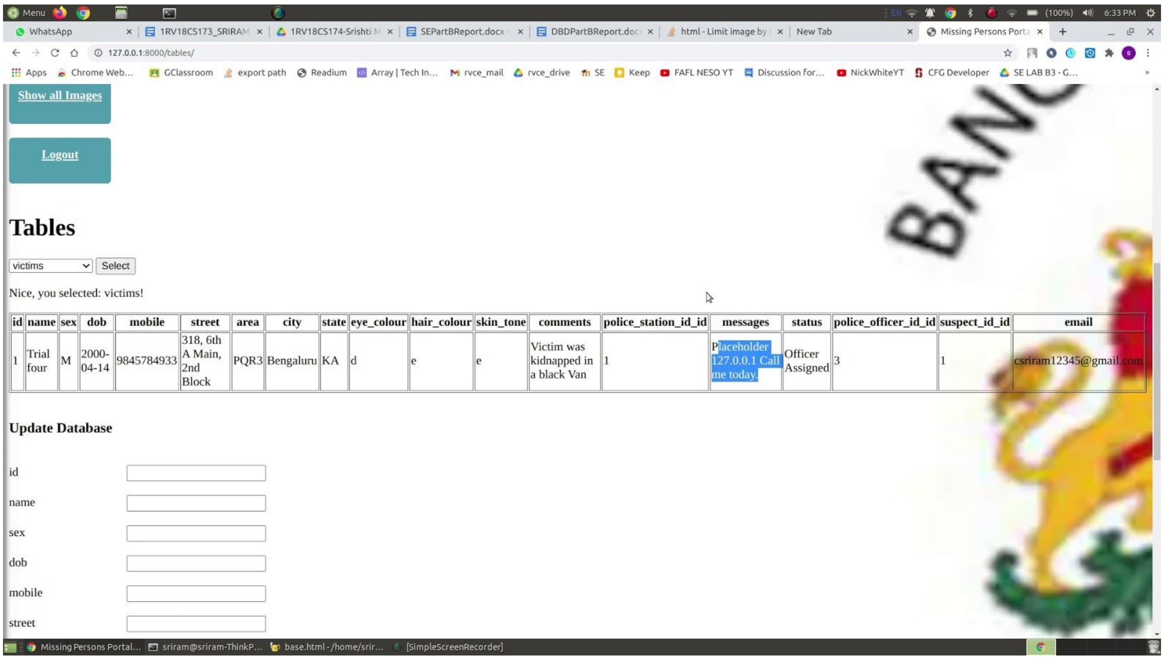
Trial four 9845784933

Enter secure key sent to your email:

Enter any message you wish to convey to the police officer:

Submit

Fig B.8 Getting status and sending a message to officer.



Tables

victims Select

Nice, you selected: victims!

id	name	sex	dob	mobile	street	area	city	state	eye_colour	hair_colour	skin_tone	comments	police_station_id_id	messages	status	police_officer_id_id	suspect_id_id	email
1	Trial four	M	2000-04-14	9845784933	318, 6th A Main, 2nd Block	PQR3	Bengaluru	KA	d	e	e	Victim was kidnapped in a black Van	1	Placeholder 127.0.0.1 Call me today.	Officer Assigned	3	1	csriram12345@gmail.com

Update Database

id

name

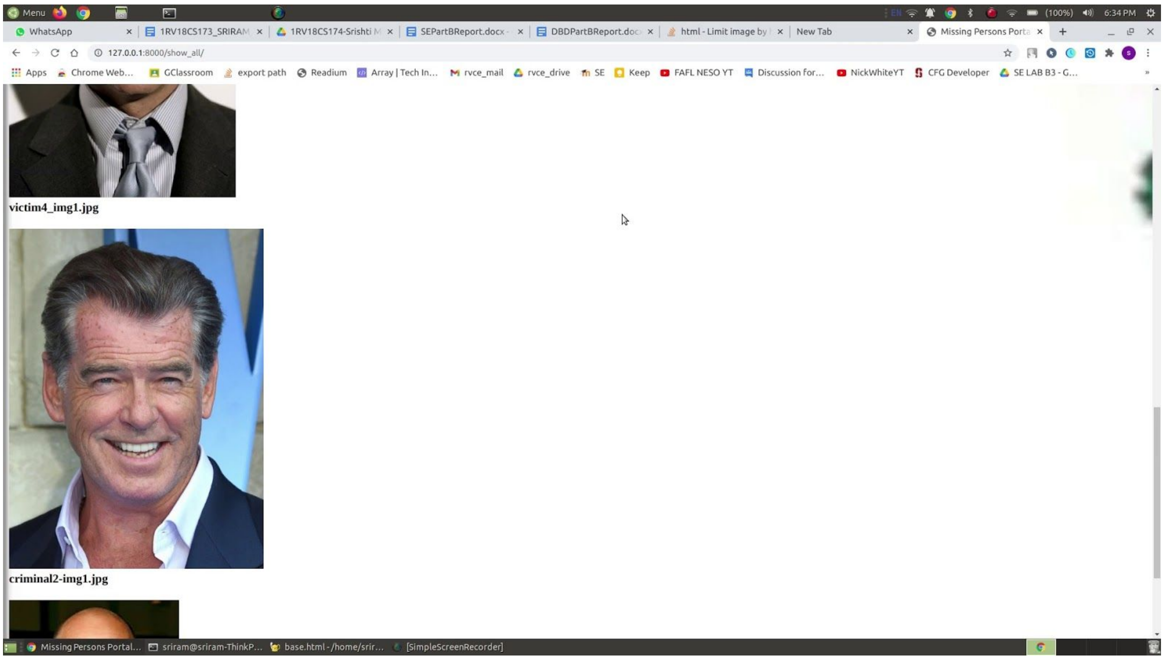
sex

dob

mobile

street

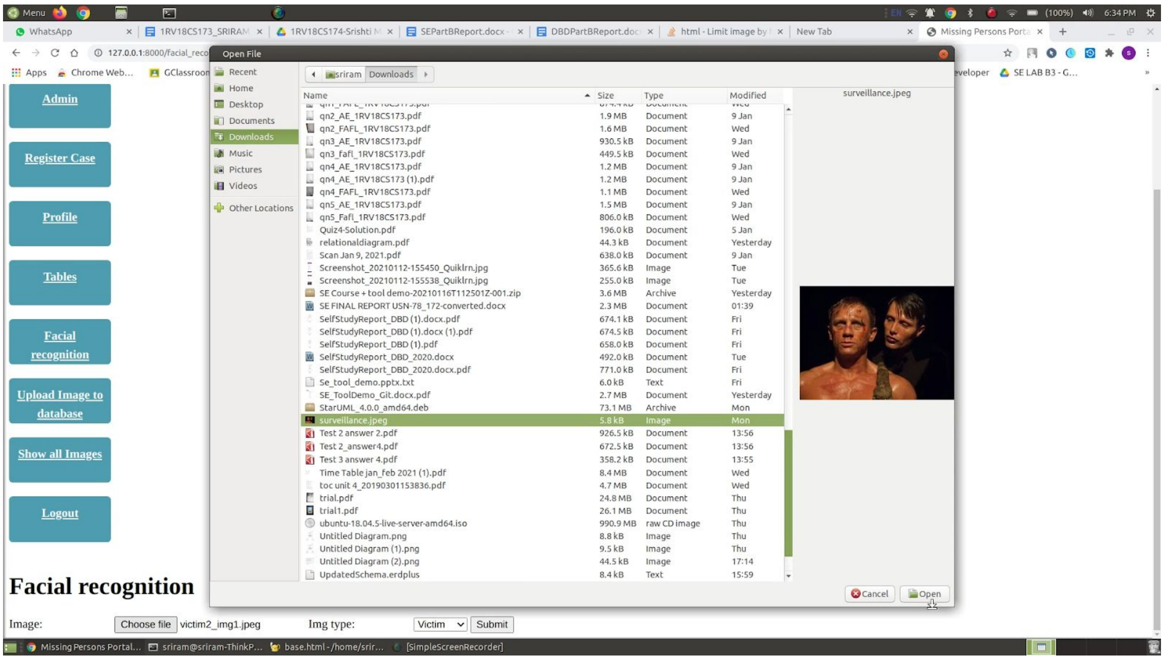
Fig B.9 Message received by officer with victim’s IP address.



victim4_img1.jpg

criminal2-img1.jpg

Fig B.10 All images in MongoDB (See Fig 6.2) displayed with the filename.

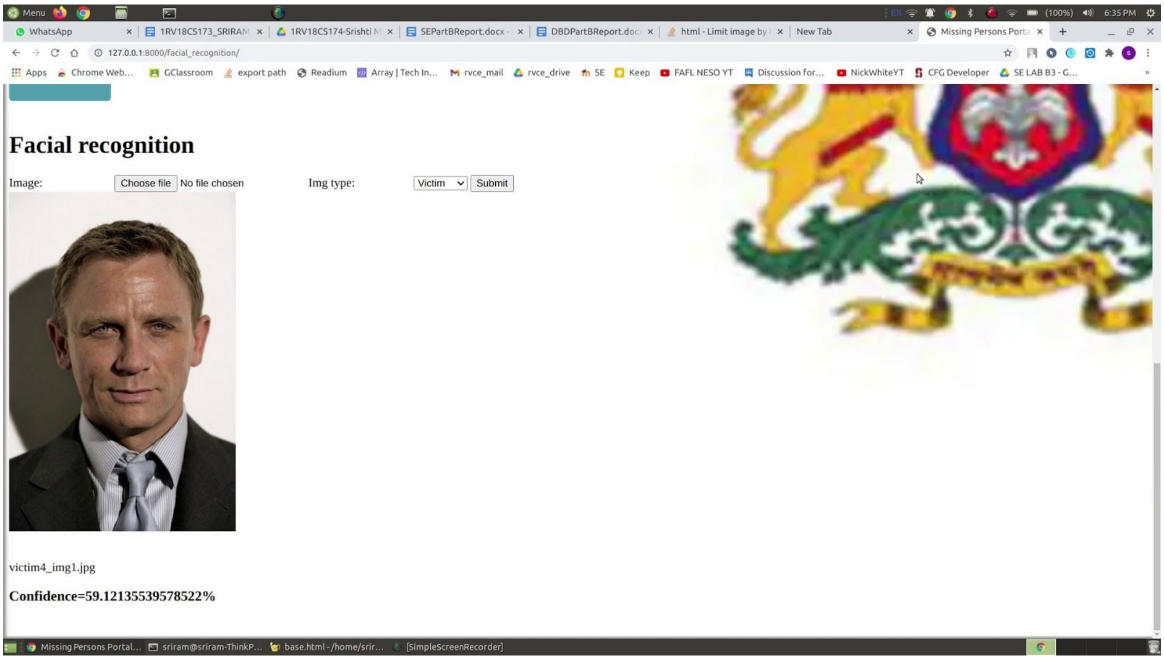


Name	Size	Type	Modified
qn2_AE_1RV18CS173.pdf	1.9 MB	Document	9 Jan
qn2_FAFL_1RV18CS173.pdf	1.6 MB	Document	Wed
qn3_AE_1RV18CS173.pdf	930.5 kB	Document	9 Jan
qn3_fafl_1RV18CS173.pdf	449.5 kB	Document	Wed
qn4_AE_1RV18CS173.pdf	1.2 MB	Document	9 Jan
qn4_AE_1RV18CS173 (1).pdf	1.2 MB	Document	9 Jan
qn4_FAFL_1RV18CS173.pdf	1.1 MB	Document	Wed
qn5_AE_1RV18CS173.pdf	1.5 MB	Document	9 Jan
qn5_Fafl_1RV18CS173.pdf	806.0 kB	Document	Wed
Quiz4-Solution.pdf	196.0 kB	Document	5 Jan
relationaldiagram.pdf	44.3 kB	Document	Yesterday
Scan Jan 9, 2021.pdf	638.0 kB	Document	9 Jan
Screenshot_20210112-155450_Quiklrn.jpg	365.6 kB	Image	Tue
Screenshot_20210112-155538_Quiklrn.jpg	255.0 kB	Image	Tue
SE Course + tool demo-20210116T112501Z-001.zip	3.6 MB	Archive	Yesterday
SE FINAL REPORT USN-78_172-converted.docx	2.3 MB	Document	01:39
SelfStudyReport_DBD (1).docx.pdf	674.1 kB	Document	Fri
SelfStudyReport_DBD (1).docx (1).pdf	674.5 kB	Document	Fri
SelfStudyReport_DBD (1).pdf	658.0 kB	Document	Fri
SelfStudyReport_DBD_2020.docx	492.0 kB	Document	Tue
SelfStudyReport_DBD_2020.docx.pdf	771.0 kB	Document	Fri
Se_tool_demo.pptx.txt	6.0 kB	Text	Fri
SE_ToolDemo_Git.docx.pdf	2.7 MB	Document	Yesterday
StarUML_4.0.0_amd64.deb	73.1 MB	Archive	Mon
surveillance.jpeg	5.8 kB	Image	Mon
Test 2 answer 2.pdf	926.5 kB	Document	13:56
Test 2_answer4.pdf	672.5 kB	Document	13:56
Test 3 answer 4.pdf	358.2 kB	Document	13:55
Time Table jan_feb 2021 (1).pdf	8.4 MB	Document	Wed
toc unit 4_20190301153836.pdf	4.7 MB	Document	Wed
trial.pdf	24.8 MB	Document	Thu
trial1.pdf	26.1 MB	Document	Thu
ubuntu-18.04.5-live-server-amd64.iso	990.9 MB	raw CD image	Thu
Untitled Diagram.png	8.8 kB	Image	Thu
Untitled Diagram (1).png	9.5 kB	Image	Thu
Untitled Diagram (2).png	44.5 kB	Image	17:14
UpdatedSchema.erdplus	8.4 kB	Text	15:59

Facial recognition

Image: Choose file victim2_img1.jpeg Img type: Victim Submit

Fig B.11 Facial recognition image chosen



Facial recognition

Image: Choose file No file chosen Img type: Victim Submit

victim4_img1.jpg

Confidence=59.12135539578522%

Fig B.12 Closest image returned with confidence score