INTRODUCTION

1.1 PROJECT PROFILE

It is a web-based application software in which could login to the website.People can report crime online. It provides the facility of uploading images of crime scenes to ensure that police may take action immediately. It also supplies the advice of missing persons, most wanted criminals and security tips for the awarness of people.This project has been proposed to reduce manual work, improving work efficiency, saving time and to provide greater flexibility and user-friendliness.

1.2 ORGANIZATION OVERVIEW

KITES Software's Pvt. Ltd, is one of the fastest growing I.T Company with a vision of achieving new innovative heights in the field of Software Development, Consultancy, Services and Training.

In a short span of time, we have established our credentials through our unique, highly customized service model that have enabled our clients reduce cost substantially, shorten lead times for processes, improve financial reporting and focus on their core businesses more efficiently.

At KITES, we continuously strive to add and fine-tune our knowledge base in order to harness newer cutting-edge technologies to our services. It is a matter of pride that nationally reputed scientific institutions like NIT Calicut have placed their faith in our abilities & provided us with all managerial & technical assistance for projects under the Technology Business Incubator (TBI) program.

SYSTEM SPECIFICATION

2.1 RECOMMENDED SOFTWARE SPECIFICATION

Operating System	Window Server 2018 & Higher, Window 10 pro or higher
Runtime Framework	Django 3.2
Web server	IIS 7.0
Front-End	Python 3.9.4
Back-End	SQLite3
Web browser	Microsoft Edge or any compatible browser.

2.2RECOMMENDED HARDWARE SPECIFICATION

Processor	3.50 GHz or Higher
RAM	8.00 GB
Hard Disk	1 TB
Mouse	Standard Mouse
Keyboard	Logitech Keyboard

DEVELOPING TOOL

3.1 FRONT-END

Python

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very effective.

3.2 BACK-END

SQLite3

SQLite is a relational database management system (RDBMS) contained in a C library. In contrast to many other database management systems, SQLite is not a client–server database engine. Rather, it is embedded into the end program. SQLite generally follows PostgreSQL syntax. SQLite uses a dynamically and weakly typed SQL syntax that does not guarantee the domain integrity. This means that one can, for example, insert a string into a column defined as an integer. SQLite will attempt to convert data between formats where appropriate, the string "123" into an integer in this case, but does not guarantee such conversions and will store the data as-is if such a conversion is not possible. SQLite is a popular choice as embedded database software for local/client storage in application software such as web browsers. It is arguably the most widely deployed database engine, as it is used today by several wide spread browsers,

operating systems, and embedded systems (such as mobile phones), among others. SQLite has bindings to many programming languages.

Features of SQLite

The following lists how's the most important properties of SQLite.

- SQLite is totally free:

SQLite is open-source. So, no license is required to work with it.

- SQLite is server less:

SQLite doesn't require a different server processor system to operate.

- SQLite is very flexible:

It facilitates you to work on multiple data bases on the same session on the same time.

- Configuration Not Required:

SQLite doesn't require configuration. No setup or administration required.

- SQLite is a cross-platform DBMS:

You don't need a large range of different platforms like Windows, MacOS, Linux, and Unix. It can also be used on a lot of embedded operating systems like Symbian, and Windows CE.

- Storing data is easy:

SQLite provides an efficient way to store data.

- Variable length of columns:

The length of the columns is variable and is not fixed. It facilitates you to allocate only the space a field needs. For example, if you have a varchar (200) column, and you put a10 characters' length value on it, then SQLite will allocate only 20 characters' space for that value not the whole 200 space.

- Provide large number of API's:

SQLite provides API for a large range of programming languages. For example: .Net languages (Visual Basic, C#), PHP, Java, Objective C, Python and a lot of other programming language. SQLite is written in ANSI-C and provides simple and easy-to-use API. SQLite is available on UNIX (Linux, Mac OS-X, Android, iOS) and Windows (Win32, WinCE, WinRT).

Advantages of SQLite

There are a lot of advantages to use SQLite as an application file format:

- Lightweight:

SQLite is a very light weighted database so, it is easy to use it as an embedded software with devices like televisions, Mobile phones, cameras, home electronic devices, etc.

- Better Performance:

Reading and writing operations are very fast for SQLite database. It is almost 35%faster than Filesystem. It only loads the data which is needed, rather than reading the entire file and hold it in memory. If you edit small parts, it only over writes the parts of the file which was changed.

- No Installation Needed:

SQLite is very easy to learn. You don’t need to install and configure it. Just download SQLite libraries in your computer and it is ready for creating the database.

1. Reliable:

It updates your content continuously so, little or no work is lost in a case of power failure or crash. SQLite is less bugs prone rather than custom written file I/O codes. SQLite queries are smaller than equivalent procedural codes so, chances of bugs are minimal.

2. Portable:

SQLite is portable across all 32-bit and 64-bit operating systems and big- and little-endian architectures. Multiple processes can be attached with same application file and can read and write without interfering each other. It can be used with all programming languages without any compatibility issue.

3. Accessible:

SQLite database is accessible through a wide variety of third-party tools. SQLite database's content is more likely to be recoverable if it has been lost. Data lives longer than code.

4. Reduce, Cost and Complexity:

It reduces application cost because content can be accessed and updated using concise SQL queries instead of lengthy and error-prone procedural queries. SQLite can be easily extended in in future releases just by adding new tables and/or columns. It also preserves the backwards compatibility

3.3 OPERATING SYSTEM

Windows 10

Windows 10 is a series of operating systems developed by Microsoft and released as part of its Windows NT family of operating systems.

It is the successor to Windows 8.1, released nearly two years earlier, and was released to manufacturingonJuly15, 2015, and broadly released for the general public on July 29, 2015.

Windows 10 receives new builds on an ongoing basis, which are available at no additional cost to users, in addition to additional test builds of Windows 10, which are available to Windows Insiders. Devices in enterprise environments can receive these updates at a slower pace, or use long-term support milestones that only receive critical updates, such as security patches, over their ten-year life span of extended support. The Windows user interface was revised to handle transitions between a mouse-oriented interface and a touchscreen-optimized interface based on available input devices particularly on 2-in-1 PCs, both interfaces include an updated Start menu which incorporates elements of Windows 7's traditional Start menu with the tiles of Windows 8. Windows 10 also introduced the Microsoft Edge web browser, a virtual desktop system, a window and desktop management feature called Task View, support for fingerprint and face recognition login, new security features for enterprise environments, and DirectX 12.

Features of Windows10

Windows10 introduced several new features to the Windows line, including:

Cortana is a search function first implemented on Windows Phone 8.1. It uses your personal information in conjunction with Bing-powered cloud smarts to find info you're looking for. Cortana can schedule a reminder, set alarms, check your calendar, call a contact, compare stocks, and much more. It responds to either typed or voice commands.

Windows 10 gives you one easy-to-navigate Settings menu, available as soon as you click the Start button. They've done away with the strangely designed separate Settings menus of Windows 8.

Windows 10 has been formatted to better fit touchscreen devices. The Continuum feature allows users to switch between desktop mode and a style similar to Windows 8 built for mobile devices. Hybrid devices will alternate between both modes depending on if the user has attached a keyboard. While in tablet mode the Start menu expands to fit the full screen. The Action Center has a button called Tablet Mode which will allow you to switch the feature off or on.

Start Page is gone, and with Windows10 you are launched automatically to desktop.

Microsoft Edge is the new browser for Windows 10 and is the successor to Internet Explorer11, although Internet Explorer will remain for compatibility and legacy purposes. Cortana has been integrated into Edge, accessible by the option "Ask Cortana" in the right click menu, as well as a Reading View and the ability to write notes directly on web pages and save to OneNote.

Windows 10 introduced Universal Windows Platform (UWP), an extension of the Windows Runtime platform which was originally introduced with Windows 8. UWP emphasizes a core set of APIs common to all variations of the operating system, enabling the ability to code a single application with adaptations for different device families and states, including desktops and laptops, tablets, smartphones (via Windows 10 Mobile), Xbox One, and other new device classes such as Surface Hub and HoloLens.

.

SYSTEM ANALYSIS

4.1 INTRODUCTION

By conducting the problem definition find out what are the objectives and goals of the proposed system, whether they are satisfied or not, what are the requirements and what are the advantages and disadvantages of the existing system. The structure also explains the kind of environment and what happens to data through various processes in the system.

Training, experience and common sense are required for the collection of the information needed to do the system analysis involves studying the ways an organization currently retrieves and process data to produce information with the goal of determining how to make it work better. To do so, the system analyst may develop alternate systems and evaluate in terms of cost benefit and feasibility.

In the existing system most of the functions have been done manually, which have some drawbacks. Manual processing is tedious and time consuming. The staffs are unable to deal with number of operations effectively.

By computerizing these functions can be done quickly and we can avoid unnecessary time wastage. This system is automated considering the labor needed in the completion of the day-to-day calculations done on that information. Storage of details can be achieved through this system.

Since the automation is preferred to manual labor, automating of an existing system will certainly lead to a better-integrated working environment, efficient handling of data and meaningful interaction with the information.

4.2 EXISTING SYSTEM

In present system all work is done on paper. The existing system was a manual one. Whatever be the process involved in the system were done through register (files). There were lots of complexities involved in the system. Presently there is no online web application available to report crime online. In order to report any complains associated with crime, people has to contact closest police station. People of the specific city are not aware of crime related matters such as list of Most wanted criminals of their town, newest crime related news, missing persons of the area etc.. People must view News Channels or Read News Paper for such offense related details. Thus we can say that present platform is manual and does not supply all the information from 1 source.

4.3 LIMITATION OF EXISTING SYSTEM

The justifications for a new candidate system is due to some of the disadvantages faced by the existing system are listed below:

- Not user friendly.
- Difficulty crime reporting.
- Manual control.
- Lost of paperwork.
- Time consuming.
- Traveling expenses

The existing manual system is facing this much problems. While using the existing system we are losing our valuable time, effort, cost and human resources.

4.3 FEASIBILITY STUDY

During the system analysis a feasibility study of proposed system carried out to see whether it was beneficial to the organization. Three key considerations are involved and results are:

4.3.1 TECHNICAL FEASIBILITY

Around existing environment and to what extend it can support the proposed system. While considering the technical factors of the organization that it presently have is sufficient to implementation of the new system. The new system can use the existing premises inside the firm and no extra premises needed. In my project of "**Crime Reporting**" technical feasibility has been conducted successfully. It is found that this project is technically feasible.

4.3.2 ECONOMICAL FEASIBILITY

Most frequently used as evaluating the effectiveness of candidate system more commonly known as cost benefit analysis. The procedure is to determine the benefits and saving that are expected from a candidate system and compare them with the existing system. If the benefits of the candidate system are out of weight the existing, the decision is made to design and implement. In my project of "**Crime Reporting**" economic feasibility has been conducted successfully. It is found that this project is economically feasible.

4.3.3 OPERATIONAL FEASIBILITY

People are inherently resisted to change and computers have been known to facilitate change. This system can be implemented in the organization because there is adequate support from management and users. In my project of "**Crime Reporting**" operational feasibility has been conducted successfully. It is found that this project is operationally feasible

4.4 PROPOSED SYSTEM

The graphical user interface is provided in the proposed system. Which provide user to deal with the system very easily. All the data is feted into the computer immediately and reports can be generated through computers. The proposed and thereby upgraded system of **Crime Reporting** provides the customer to be completely free of physical work where the customer can stay at home or be at any place with internet availability.

4.5 ADVANTAGES OF PROPOSED SYSTEM

The advantages of proposed system are:

- User friendly.
- Reports are easily generated.
- Very less paper work.
- Computer operator control.
- Flexibility.
- Cost effective.
- Intelligent validation for each entry.
- With the help of GUI, it is user friendlier.

SYSTEM DESIGN

5.1 INTRODUCTION

System design is the solution to the creation of a new system. This phase is composed of several systems. This phase focuses on the detailed implementation of the feasible system. System design has two phases of development logical and physical design. During logical design phase the analyst describes inputs (sources), outputs (destinations), databases (data source) and procedures (data flows) all in a format that meets the user requirements.

Design goes through the logical and physical stages of development. At an early stage in designing a new system, the system analyst must have a clear understanding of the objectives, which the design is aiming to fulfill. Second input data and master files (database) have to be designed to meet the requirements of the proposed output. The operational (processing) phases are handled through program construction and testing.

The system design includes:

- Output design
- Database design
- Input design
- Modularization

5.2 OUTPUT DESIGN

Computer output is the most important and direct information source to the user. Output design is a process that involves designing necessary outputs in the form of reports that should be given to the users according to the requirements. Efficient, intelligible output design should improve the system's relationship with the user and help in decision making.

So while designing output the following things are to be considered:

- Determine what information to present.
- Arrange the presentation of information in an acceptable format.
- Decide how to distribute the output to intended receipts.

5.3 DATABASE DESIGN

Database design is the logical form of design of data storage in the form of records in a particular structure in the form of tables with fields which is not transparent to the normal user but it actually acts as the backbone of the system. As we know database is a collection which helps the system to manage and store data is called database management system. Data base management system builds some form of constraints like integrity constraints, i.e., the primary key or unique key and referential integrity which help to keep data structure storage and access of data from tables efficiently and accurately and take necessary steps to concurrent access of data and avoid redundancy of data in tables by normalization criteria. Normalization is the method of breaking down complex table structures into simple table structures by using certain rules, thus reduce redundancy and inconsistency and disk space usage and increase the performance of the system or application which is directly linked to the database design and also solve the problems of anomalies.

The data base design of the new system is in third normal form and every normalization are:

- First normal form (1NF)
- Second normal form (2NF)
- Third normal form (3NF)
- Boyce code normal form
- Forth normal form (4NF)
- Fifth normal form (5NF)

First normal form (1NF) sets the fundamental rules for database normalization and relates to a single table within a relational database system. Normalization follows three basic steps, each building on the last. The first of these is the first normal form

The first normal form states that:

- Every column in the table must be unique
- Separate tables must be created for each set of related data
- Each table must be identified with a unique column or concatenated columns called the primary key
- No rows may be duplicated

First Normal Form (1NF)

The first step in confirming 1NF is modifying multivalued columns to make sure that each column in a table does not take more than one entry. Searching records with duplicate entries is complex. To overcome this situation, all records involved in a relational database table have to be identified by a unique value which will have a separate column (or attribute). This unique key is called an index key and is used to locate data for retrieval or other manipulation.

Having a unique key does not resolve the requirements of 1NF. According to the rules, there can be no multiple entries into a single field. For example, in a data table of customer information, a single field could be allowed to store multiple entries, such as where a customer has multiple telephone numbers. This is a violation of the 1NF rules. This particular problem in our example can be resolved by creating a customer ID index in the main table and then adding a separate table that has a column for the telephone numbers and another column for the customer ID. This allows proper use of relational queries to extract data from a relational database. Null, or multiple entry fields both cause issues with data manipulation and extraction so the normalizing process removes ambiguity.

Removing repeating values from a table is the next step toward first normalized form. Repeating values can be moved to a new table.

The final step in implementing first normal form is maintaining atomicity of data. Each individual field should hold the smallest data element possible to facilitate easy sorting and searching. For instance, the date column can be separated into day, month and year. Tables satisfying first normal form can also contain functionally dependent fields. Functional dependency exists between two fields when the value in field 1 determines the value in field 2 and there is only one value in field 2. In such a case, field 2 is functionally dependent on field 1. Tables satisfying the higher normal forms (second, third and fourth) necessarily follow first normal form but the reverse is not true. All tables complying with first normal form may not follow the higher normal forms, as the higher normal forms include even more stringent rules.

Second Normal Form (2NF)

After meeting the requirements of 1NF, 2NF requires the database designer to do the following:

1. Split up all data resulting in many-to-many relationships and store the data as separate tables. For example, in a database used by a school's application, two of the tables are STUDENT and SUBJECT. In real life, a student takes several subjects simultaneously while a subject is studied by several students. These are many-to-many relationships.

2NF states that this relationship must be split into more than the two tables above (STUDENT and SUBJECT). One way of splitting them is by introducing a third table, which contains the columns Student_ID, Subject_ID, Semester and Year. In this way, there is no direct relationship between STUDENT and SUBJECT because all relationships are created indirectly through the third table.

2. Create relationships between tables by use of foreign keys. For example, a bank's database contains two tables: CUSTOMER_MASTER (for storing customer details) and ACCOUNT_MASTER (for storing details about bank accounts, including which customer holds which account). There must be a way to link the two tables to know who the customer is for each account. The way to do this is via a foreign key, which is a column in the ACCOUNT_MASTER table pointing to a corresponding column in the CUSTOMER_MASTER table.

A table for which there are no partial functional dependencies on the primary key might or might not be in 2NF. In addition to the primary key, the table may contain other candidate keys; it is necessary to establish that no non-prime attributes have part-key dependencies on any of these candidate keys.

Third Normal Form (3NF)

These three "normal forms" or normalizations tend to confound people who aren't professional database administrators or mathematicians.

However, here's an easy and simple way to think about third normal form, as well as the two normal forms that precede it.

As you might imagine, much like dimensional complexity, the first normal form is very basic compared to the forms that come after it. That's also true for the second normal form as well.

Here's the important part: no transitive dependency for non-prime attributes.

- Also, in a 3NF-compliant table, no non-primary key attribute has transitively dependent relationships to the primary key.

- Again, this has to do with the relations between items in a database table, but it's more complex. Here's an easy way to think about third normal form — it makes sure that these fields will not have anomalies based on changes — administrative inserts, updates, and deletions.

- So in general, it's the process of engineering your data table the right way, so that every value has independence, and your procedural changes don't damage other parts of your data table as you execute them.
- That's something that's fairly easy to understand when you're looking at using candidate keys and primary keys to engineer databases this way.
- You can also understand the three cascading normal forms in relation to each other — that normalization proceeds according to these steps.
- Maybe a system is compliant with first normal form, but not the other two.
- However, it won't be compliant only with second or third normal form, because of the prerequisite nature of the set.
- So that's it in a nutshell — again, 3NF means that various parts of the record are independent, so that changes don't cause unintended consequences.

Boyce-Codd Normal Form (BCNF)

BCNF was developed by Raymond Boyce and E.F. Codd; the latter is widely considered the father of relational database design.

BCNF is really an extension of 3rd Normal Form (3NF). For this reason it is frequently termed 3.5NF. 3NF states that all data in a table must depend only on that table's primary key, and not on any other field in the table. At first glance it would seem that BCNF and 3NF are the same thing. However, in some rare cases it does happen that a 3NF table is not BCNF-compliant. This may happen in tables with two or more overlapping composite candidate keys.

Fourth Normal Form (4NF)

A multivalued dependency is best illustrated using an example. In a table containing a list of three things - college courses, the lecturer in charge of each course and the recommended book for each course - these three elements (course, lecturer and book) are independent of one another. Changing the course's recommended book, for instance, has no effect on the course itself. This is an example of multivalued dependency: An item depends on more than one value. Thus, 4NF states that a table should not have more than one of these dependencies. 4NF is rarely used outside of academic circles.

5.4 TABLE DESIGN

5.4.1 Customer side tables

Table Name: user_login

Table Description: Login details of user.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
UserId	Integer (10)	Primary key	The id of the user
Username	Varchar (20)	Not Null	Name of the user
Password	Varchar (20)	Not Null	Password of the user
Usertype	Varchar (50)	Not Null	Type of user

Table Name: User_Details

Table Description: User details storing table.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
UserId	Integer (10)	Primary key	Id of the user
FirstName	Varchar (50)	Not Null	First Name of the user
LastName	Integer (10)	Not Null	Last Name of the user
Gender	Varchar (50)	Not Null	Gender of the user
DateOfBirth	Integer (10)	Not Null	Date of birth of user
Aadhaarno	Integer(10)	Not null	Aadhaarno of user
Address	Varchar (100)	Not Null	Address of the user
PinCode	Integer (10)	Not Null	Pin code of user
ContactNumber	Integer (10)	Not Null	Contact number of theuser
Email	Varchar (50)	Not Null	Email id of user
status	Varchar(50)	Not Null	Status of user

Table Name: Crime reporting

Table Description: Crime Report storing table.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
report_id	Integer (8)	Primary key	Crime report id
user_id	Integer(8)	Foreign Key	Id of User Referred from User Table
station_id	Integer(8)	Foreign Key	Id of station Referred from Station Table
type_id	Integer(8)	Not null	Id of crime type
description	Varchar(100)	Not null	Description of crime
addr	Varchar(50)	Not null	Exact spot
place_id	Integer(8)	Foreign Key	Id of place Referred from Place Table
Dt	Date	Not null	Date
tm	Time	Not null	Time

Table Name: Crime reporting Pic

Table Description: Crime Report Pic storing table.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
report_pic_id	Integer (8)	Primary key	Crime report id
report_id	Integer(8)	Foreign Key	Id of report Referred from crime reporting Table
Pic	Varchar(100)	Not null	Pic of crime
pic_info	Varchar(100)	Not null	Pic information

Table Name: feedback

Table Description: Feedback storing table.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
feed_id	Integer (8)	Primary key	Feedback id
user_id	Integer(8)	Foreign Key	Id of User Referred from User Table
Msg	Varchar(60)	Not null	Message
dt	Date	Not null	Date
tm	Time	Not null	Time

Table Name: Message Police

Table Description: Message storing table.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
msg_id	Integer (8)	Primary key	Message id
user_id	Integer(8)	Foreign Key	Id of User Referred from User Table
Msg	Varchar(100)	Not null	Message
Dt	Date	Not null	Date
tm	Time	Not null	Time

5.4.2 Admin side tables

Table Name: State

Table Description: State Name storing table.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
state_id	Integer (8)	Primary key	Id of state
state_name	Varchar(20)	Not null	State name

Table Name: district

Table Description: District Name storing table.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
district_id	Integer (8)	Primary key	Id of district
state_id	Integer(8)	Foreign Key	Id of state Referred from State Table
district name	Varchar(20)	Not null	district name

Table Name: place

Table Description: Place Name storing table.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
place_id	Integer (8)	Primary key	Id of place
district_id	Integer(8)	Foreign Key	Id of district Referred from District Table
place_name	Varchar(20)	Not null	Place name

Table Name: Police station

Table Description: Police station Details storing table.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
police_station_id	Integer (8)	Primary key	Id of Police Station
station_name	Varchar(20)	Not null	Name of Station
station_type	Varchar(20)	Not null	Type of Station
station_descp	Varchar(100)	Not null	Station details
Saddr	Varchar(100)	Not null	Description of crime
addr	Varchar(30)	Not null	Station address
spin	Integer(8)	Not null	Station pin code
place_id	Integer(10)	Foreign Key	Id of place Referred from Place Table
s_contact1	Integer(10)	Not null	Station number 1
s_contact2	Integer(10)	Not null	Station number 2
Email	Varchar(50)	Not null	Email id of Station
status	Varchar(50)	Not null	Status of station

Table Name: Station user

Table Description: Police Details storing table.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
station_user_id	Integer (8)	Primary key	Id of Station officers
police_station_id	Integer(8)	Foreign Key	Id of Police Station Referred from Police Station Table
user_id	Integer(8)	Foreign Key	Id of User Referred from User Table
name	Varchar(30)	Not null	Name of officer
designation	Varchar(50)	Not null	designation
email	Varchar(50)	Not null	Email id of officer
contact	Integer(10)	Not null	Number of officer
status	Varchar(50)	Not null	Status of station

Table Name: Crime type

Table Description: Crime type storing table.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
crime_type_id	Integer (8)	Primary key	Crime followup id
type_name	Varchar(100)	Not null	Type of crime

5.4.3 Police side tables

Table Name: Crime report followups

Table Description: Crime report followups storing table.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
followup_id	Integer (8)	Primary key	Crime followup id
report_id	Integer(8)	Foreign Key	Id of report Referred from Crime Reporting Table
user_id	Integer(8)	Foreign Key	Id of User Referred from User Table
remarks	Varchar(100)	Not null	Updations
dt	Date	Not null	Date
tm	Time	Not null	Time

Table Name: Notice board

Table Description: Notice board storing table.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
notice_board_id	Integer (8)	Primary key	Notice board id
station_id	Integer(8)	Foreign Key	Id of station Referred from Station Table
title	Varchar(30)	Not null	Title
pic	Varchar(100)	Not null	image
Descp	Varchar(100)	Not null	Details
dt	Date	Not null	Date
tm	Time	Not null	Time

Table Name: Look out Notice

Table Description: Look out Notice storing table.

FIELD NAME	DATA TYPE	CONSTRAINTS	DESCRIPTION
look_out_id	Integer (8)	Primary key	Look out id
station_id	Integer(8)	Foreign Key	Id of station Referred from Station Table
name	Varchar(30)	Not null	Name of person
remarks	Varchar(100)	Not null	details
pic	Varchar(100)	Not null	Image
dt	Date	Not null	Date
Tm	Time	Not null	Time

5.5 INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data into a usable form for processing data entry. The activity of putting data into the computer for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system.

- What data should be given as input?
- How the data should be arranged or coded.
- The dialogue to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

5.6 FORM DESIGN

Admin Form: Login

This form allows the admin to login.

Admin Form: Home page

This form allows the admin see the home page

Admin Form: Add State

This form allows the admin to add State.

Admin Form: Add District

This form allows the admin to add District.

Admin Form: Add Place

This form allows the admin to add place.

Admin Form: Add Station

This form allows the admin to add Station.

Admin Form: Update Station

This form allows the admin to update Station.

Admin Form: Add Police

This form allows the admin to add police.

Admin Form: Add Crime Type

This form allows the admin to add Crime Type.

Admin Form: Change password

This form allows the admin to change his/her password

Admin Form: Logout

This form allows the admin to logout

User Form: Registration

This form allows the invalid users to get registered by providing necessary details.

User Form: Login

This form allows the valid users to login.

User Form: Home page

This form allows the user see the home page

User Form: View Look Out Notice

This form allows the user to view look out notice.

User Form: View Notice Board

This form allows the user to view notice board.

User Form: Report Case

This form allows the user to Report Case.

User Form: Follow up Case

This form allows the user to follow up Case.

User Form: Search Station

This form allows the user to Search the station.

User Form: Change password

This form allows the user to change the password.

User Form: Logout

This form allows the user to logout.

Police Form: Login

This form allows the valid police to login.

Police Form: Home page

This form allows the police see the home page

Police Form: Add Look Out Notice

This form allows the police to add look out notice.

police Form: Add Notice Board

This form allows the police to add notice board.

police Form: View Reported Case

This form allows the police to View Reported case.

police Form: Follow up Case

This form allows the police to Follow up case.

Police Form: Change password

This form allows the police to change the password.

Police Form: Logout

This form allows the police to logout.

5.7 SYSTEM MODULES

The system after careful analysis has been identified to be presented with the following modules and roles.

The modules involved are:

*Admin

*Police

*User

- **ADMIN**

The administrator is the super user of this application. Only admin have access into this admin section. Admin is position handled by the person assigned by the company. The administrator has all the information, has control all over the system.

Sub modules are:

- ♣ Managing State: In this the admin can add State, Delete state.
- ♣ Managing District: In this the admin can add District, Delete District.
- ♣ Managing Place: In this the admin can add Place, Delete Place.
- ♣ Managing Crime Type: In this the admin can add Crime Type, Delete Crime Type.
- ♣ Managing Station: In this the admin can add Station, Delete Station,Update Station.
- ♣ Managing Police: In this the admin can add Police, Delete Police.
- ♣ Change Password: In this the admin can change Password.

- **POLICE**

Sub modules are:

- ♣ Managing Notice Board: In this the admin can add Notice Board, Delete Notice Board.
- ♣ Managing Look Out Notice: In this the admin can add Look Out Notice, Delete Look Out Notice Board.
- ♣ View Reported Cases: In this the admin can view the reported cases.
- ♣ Follow Up Cases: In this the police can send the findings of reported cases.

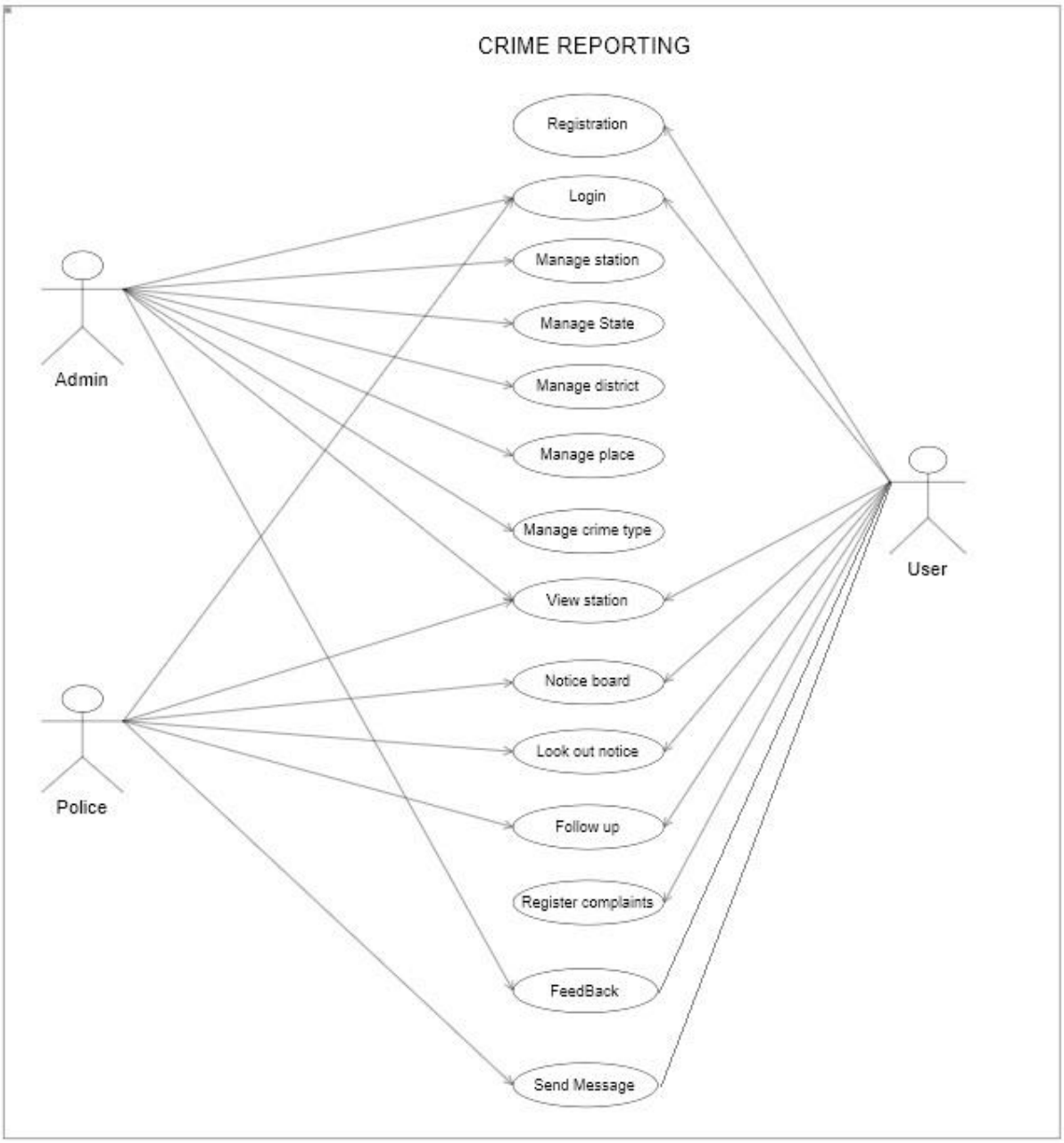
- **USER**

Sub modules are:

- ♣ Registration and Login: This module allows customer to create an account or the customer to login to their account. If the customer is registered, only they can register complaints.
- ♣ View Look Out Notice: In this the user can view Look Out Notice and download image of lookout notice.
- ♣ View Notice Board: In this the user can view Notice Board and download image of Notice Board.
- ♣ View Notice Board: In this the user can view Notice Board and download image of Notice Board.
- ♣ Register Case: In this the user can Register Case and add image of crime.
- ♣ Follow up View: In this the user can view the findings of the police of the registered case.
- ♣ Search Station: User could search the station and view the details of the station and also the Notice Board and Look Out Notice.
- ♣ Change Password: In this the User can change Password.
- ♣ Feedback: User could send feedback to admin.

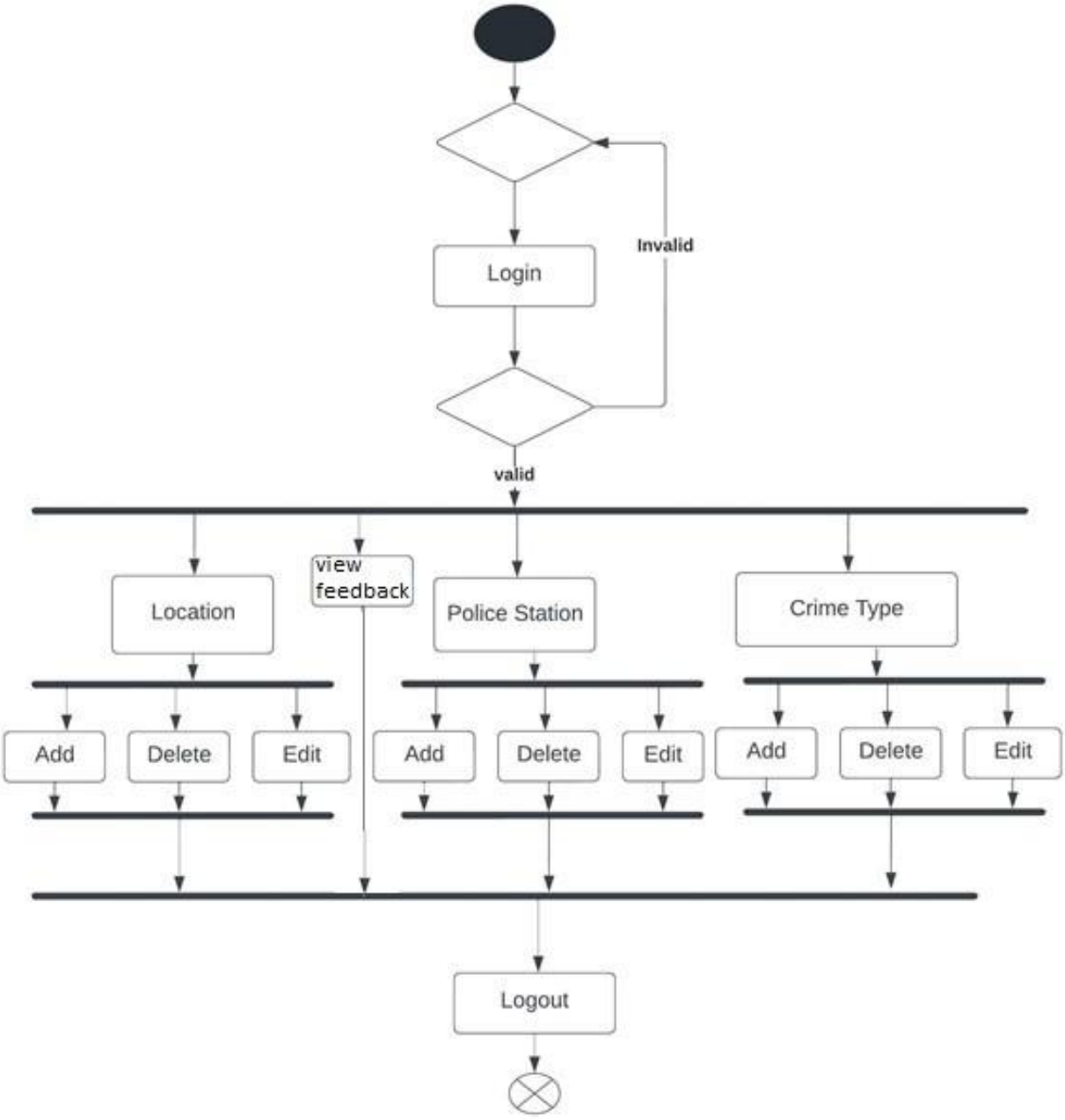
DESIGN TOOL

6.1 USE CASE DIAGRAM

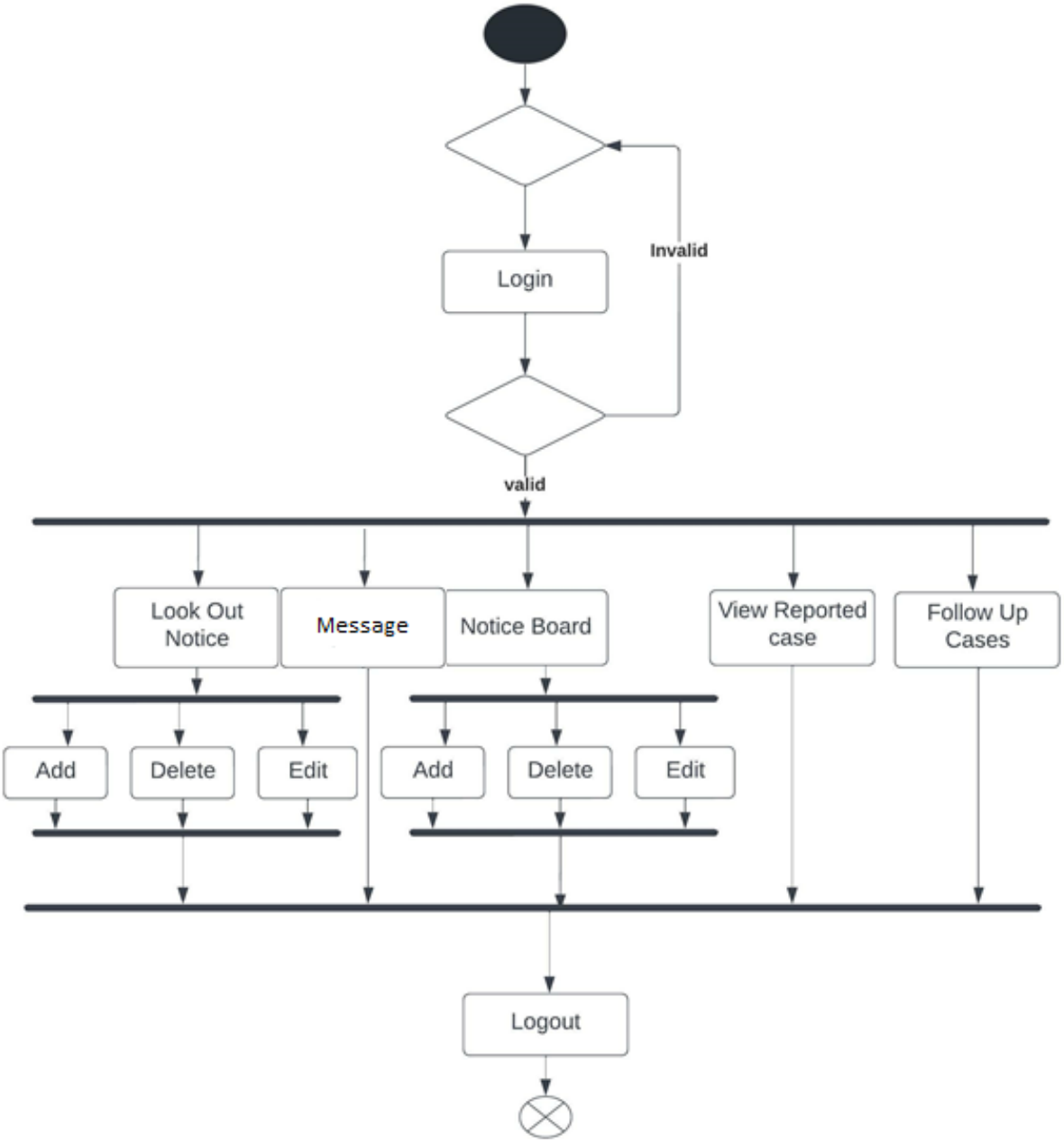


6.2 ACTIVITY DIAGRAM

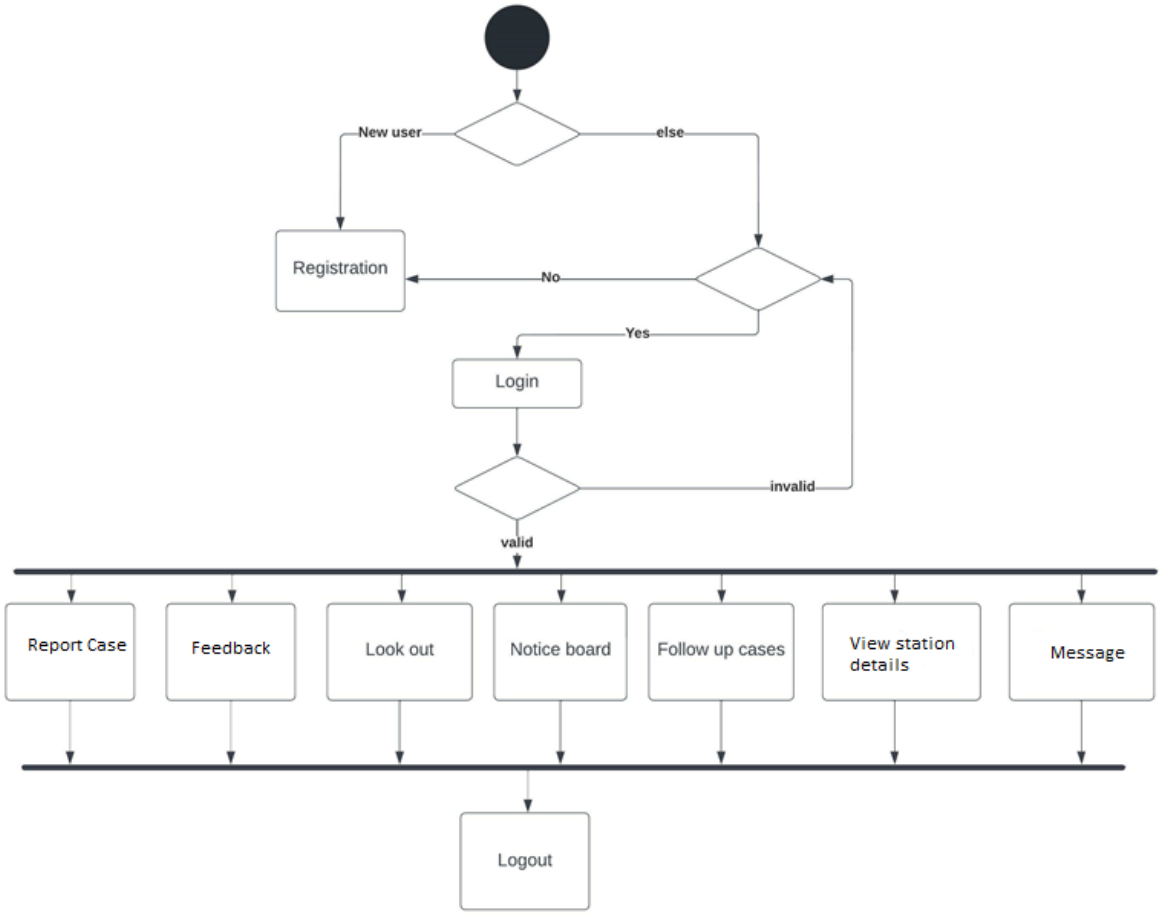
- (i)ACTIVITY DIAGRAM FOR ADMIN



- (ii) ACTIVITY DIAGRAM FOR POLICE

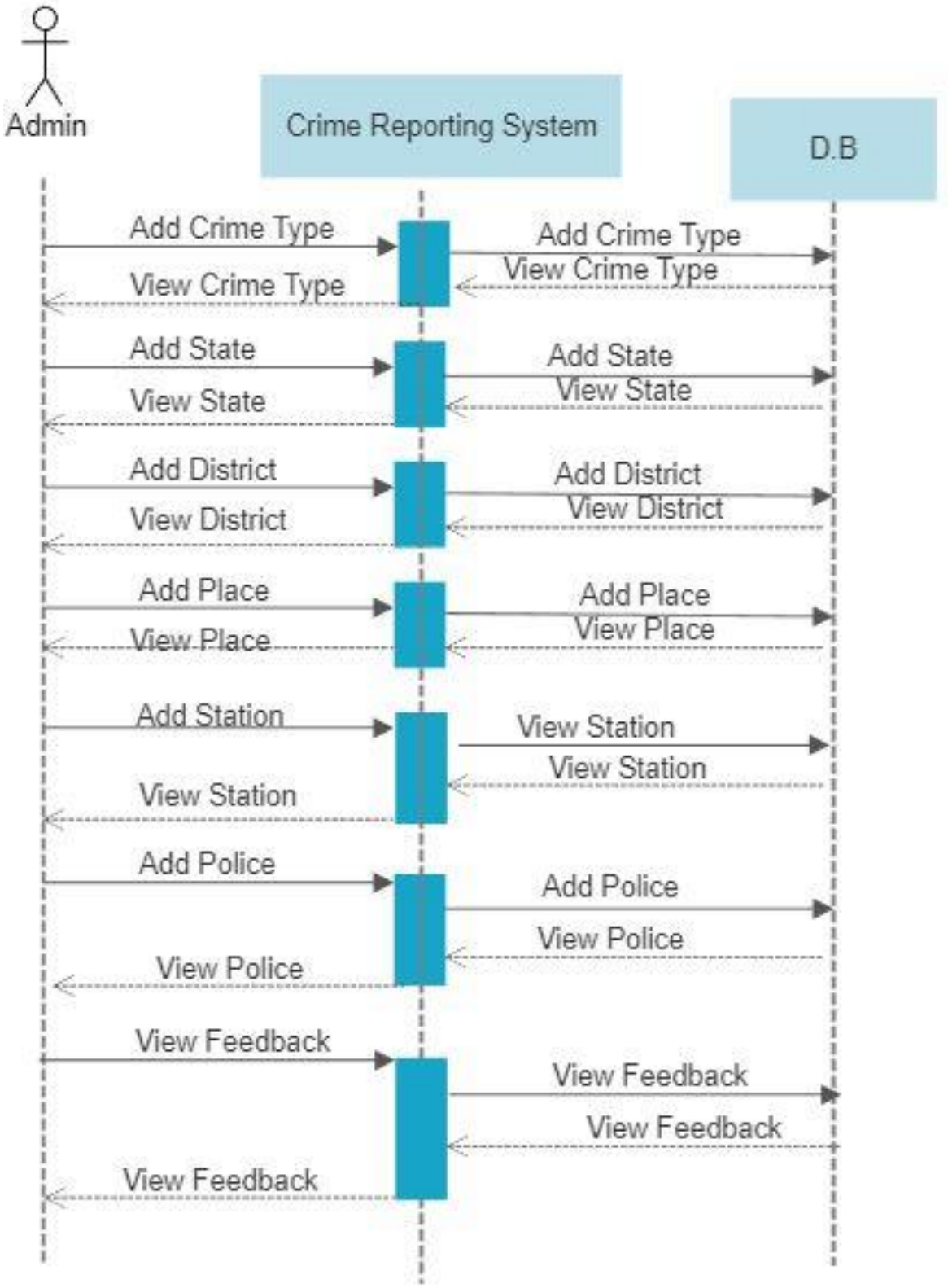


- (ii) ACTIVITY DIAGRAM FOR USER

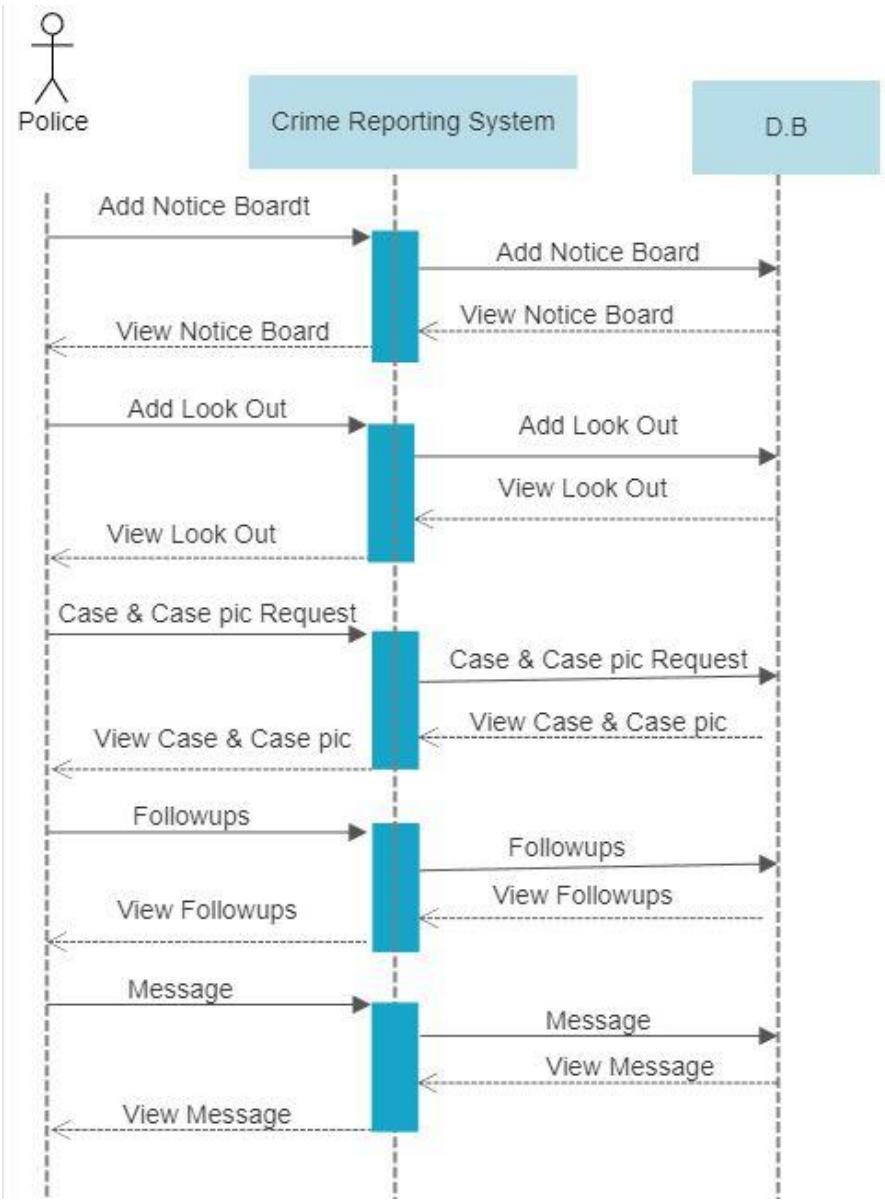


6.3 SEQUENCE DIAGRAM

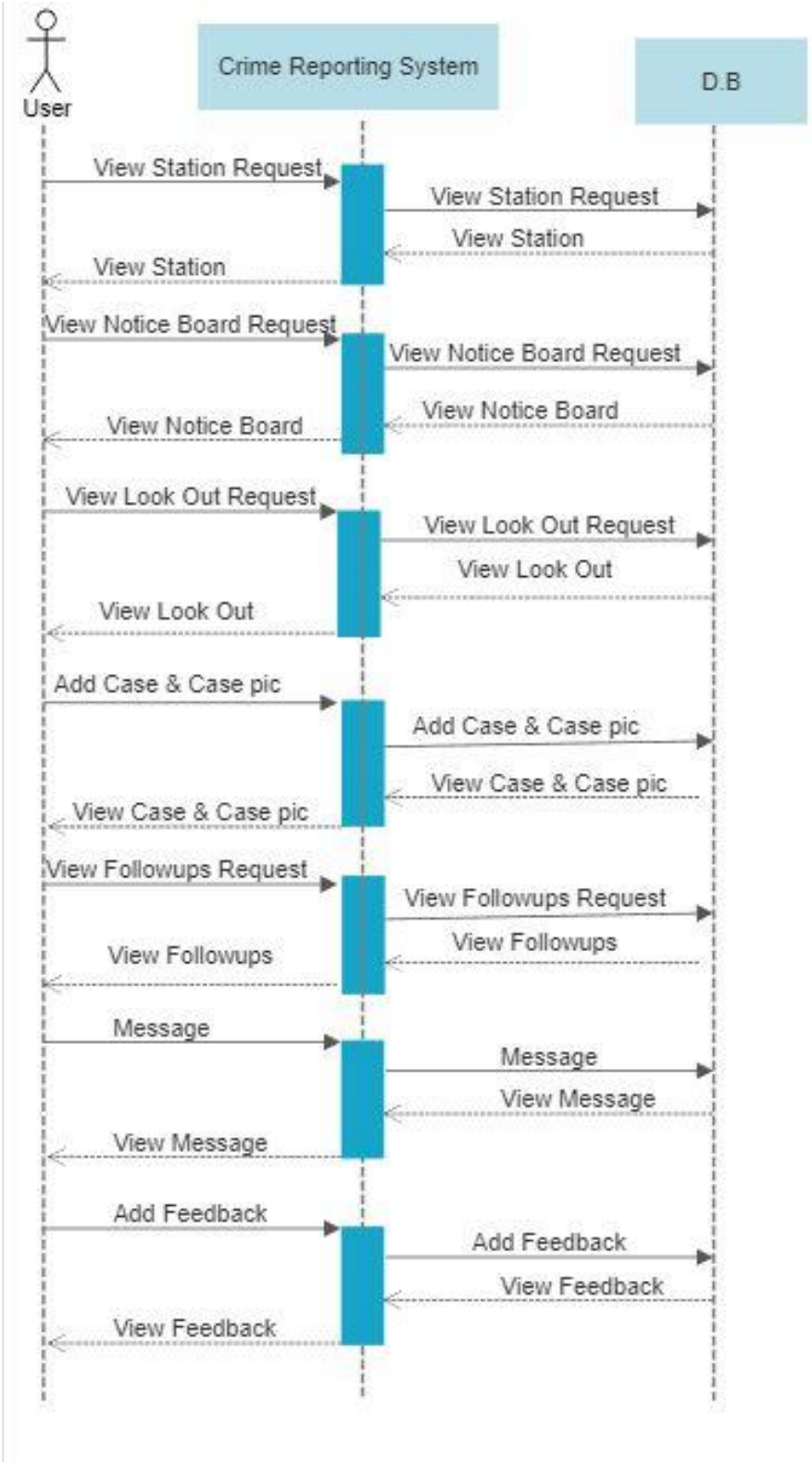
- ADMIN



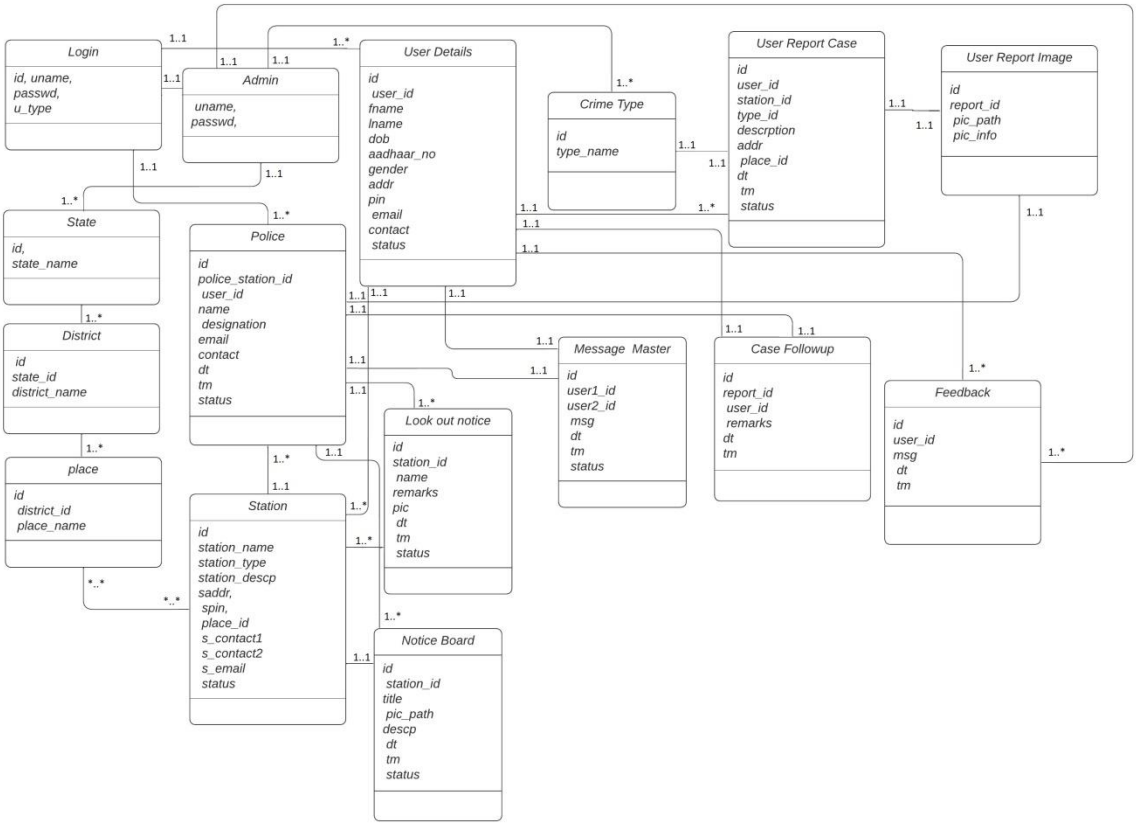
- POLICE


Police
Crime Reporting System
D.B
Add Notice Boardt
Add Notice Board
View Notice Board
View Notice Board
Add Look Out
Add Look Out
View Look Out
View Look Out
Case & Case pic Request
Case & Case pic Request
View Case & Case pic
View Case & Case pic
Followups
Followups
View Followups
View Followups
Message
Message
View Message
View Message

- USER


User
Crime Reporting System
D.B
View Station Request
View Station Request
View Station
View Station
View Notice Board Request
View Notice Board Request
View Notice Board
View Notice Board
View Look Out Request
View Look Out Request
View Look Out
View Look Out
Add Case & Case pic
Add Case & Case pic
View Case & Case pic
View Case & Case pic
View Followups Request
View Followups Request
View Followups
View Followups
Message
Message
View Message
View Message
Add Feedback
Add Feedback
View Feedback
View Feedback

6.4 CLASS DIAGRAM



SYSTEM TESTING

7.1 INTRODUCTION TO SYSTEM TESTING

System testing is a critical aspect of Software Quality Assurance and represents the ultimate review of specification, design and coding. Testing is a process of executing a program with the intent of finding an error. The purpose of testing is to identify and correct bugs in the developed system. Nothing is complete without testing. Testing is the vital to the success of the system. Testing and validation are the most important steps after the implementation of the developed system.

The system testing is performed to ensure that there are no errors in the implemented system. Validation refers to the process of using the new software for the developed system in a live environment i.e., new software inside the organization, in order to find out the errors. A specification test is conducted to check whether the specifications stating the program are performing under various conditions. Apart from these tests, there are some special tests conducted which are given below:

- PEAK LOAD TESTS

This determines whether the new system will handle the volume of activities when the system is at the peak of its processing demand. The test has revealed that the new software for the agency is capable of handling the demands at the peak time.

- STORAGE TESTING

This determines the capacity of the new system to store transaction data on a disk or on other files. The proposed software has the required storage space available, because of the use of a number of hard disks.

- PERFORMANCE TIME TESTING

This test determines the length of the time used by the system to process transaction data. In this phase the software developed Testing is exercising the software to uncover errors and ensure the system meets defined requirements.

7.2 UNIT TESTING

A Unit corresponds to a screen /form in the package. Unit testing focuses on verification of the corresponding class or Screen. This testing includes testing of control paths, interfaces, local data structures, logical decisions, boundary conditions, and error handling. Unit testing may use Test Drivers, which are control programs to co-ordinate test case inputs and outputs, and Test stubs, which replace low-level modules. A stub is a dummy subprogram.

In unit testing I've run each and every pages of the project and checked the control path whether its control flow is as per the code or not. Checked whether it responds with the expected output or not etc… likewise each and every sub modules or units are tested by running it.

7.3 INTEGRATION TESTING

Integration testing is used to verify the combining of the software modules. Integration testing addresses the issues associated with the dual problems of verification and program construction. Combine the unit tested module one by one and test the functionality of the combined unit. Normally, integration testing is carried out after unit testing. Once all the individual units are created and tested, we start combining those tested modules and start performing the integrated testing. The main goal here is to test the interfaces between the units/modules.

- Prepare the test integration plan
- Decide on the type of integration testing approach
- Design test cases, test scenarios and test scripts accordingly
- Deploy the chosen modules together and get the integration tests running
- Track the defects and record the test results of tests
- Repeat the above steps until the complete system is tested

7.4 VALIDATION TESTING

Validation succeeds when the software functions in which the user expects. Validation refers to the process of using software in a live environment in order to find errors. During the course of validating the system, failure may occur and sometimes the coding has to be changed according to the requirement.

In this online room booking project it validate each and every text fields in every pages. For example it checks for the '@' symbol in the email address field , checks the number of digits at the phone number field and make sure that it does not exceeds the limit of 10 digits. The validation also doesn't allow empty fields in any form. These are the main validations done in this project.

7.5 ALPHA TESTING

This test is the first stage of testing and will be performed amongst the developer unit testing, integration testing and system testing when combined together are known as alpha testing. For example;

- Spelling Mistakes
- Broken Links
- Cloudy Directions

7.6 BETA TESTING

This test is performed after alpha testing has been successfully performed. In beta testing a sample of the intended audience tests the application. Beta testing is also known as prerelease testing. Beta test versions of software are ideally distributed to a wide audience on the web partly

to give the program a real-world test and partly to provide a preview of the next release .In this phase the audience will be testing the following.

- Users will install, run the application and send their feedback to the present team or admin.
- Typographical users, confusing application flow, even crashes.
- Getting the feedback, the project admin can fix the problems before releasing the s/w to the actual users. The more users will fix that solve real user problems.
- Having a higher quality application when you release to the general public will increase the customer satisfaction.

7.7 TEST CASES

A Test case is a set of sequential steps to execute a test operating on a set of predefined inputs to produce certain expected outputs. There are two types of test cases; manual and automated. Mannered test case is executed manually while an automated test case is executed using automated system.

o TEST CASE FOR LOGIN

SI NO.	TEST CASE IN	TEST CASE DESCRIPTION	EXPECTED RESULT	ACTUAL RESULT	
1	Login	When no username is entered.	Display an error msg “please enter user name”.	An error msg” please enter user name”.	PASS
		When no password is entered.	Displays an error msg “please enter password”.	An error msg “please enter password”.	PASS
		When the invalid username & password is entered.	Displays an error msg “Invalid username or password”.	An error msg “Invalid username or password”.	PASS
		When the valid user name & password is entered.	Corresponding page is displayed.	Corresponding page is displayed.	PASS

o TEST CASE FOR REGISTRATION

SERAIL NO.	TEST CASE IN	TEST CASE DESCRIPTION	EXPECTED RESULT	ACTUAL RESULT	
1	Register	When any of the field is not filled.	Display an error msg "the field is required".	An error msg" The field is required".	PASS
		On clicking Register Button.	Displays login form.	Displays login form.	PASS

o TEST CASE FOR ADDING NOTICE BOARD

SERAIL NO.	TEST CASE IN	TEST CASE DESCRIPTION	EXPECTED RESULT	ACTUAL RESULT	
1	Notice Board	When any of the field is not filled.	Display an error msg "the field is required".	An error msg" The field is required".	PASS
		On clicking submit Button.	Loads the next form.	Loads the next form.	PASS

SYSTEM IMPLEMENTATION

8.1 INTRODUCTION TO SYSTEM IMPLEMENTATION

A crucial phase in the systems life cycle is the successful implementation of the new system design. Implementation simply means converting a new system design into operation. This involves creating computer compatible files, training and telecommunication network (where necessary) before the system is up and running. A critical factor in conversion is not disrupting the functioning of organization. Actual data were input into the programs and the working of the system was closely monitored. It is a process of converting a new or revised system into an operational one. It is the essential stage in achieving a successful new system because usually it involves a lot of upheaval in the user. It must there for be carefully planned and controlled to avoid problems. Apart from planning the two major tasks of preparing for implementation are education and training of the users and testing of the system. Education of the system should really have taken place much earlier in the project when they were being undelivered in the investigation and design work. The user was given necessary training for using the system. The training has made them get the effort spent on developing any system results in success only when the system is implemented properly.

The implementation phase involves the following tasks:

- Careful Planning
- Investigation of Systems and Constraints
- Design of Methods to Achieve the Changeover
- Training of the staff in the Changeover Phase
- Evaluation of Changeover.

We implemented this new system in parallel run plan without making any disruptions to ongoing system in the library section of the organization, but only computerizing the whole system to make the work, evaluation and retrieval of data easier, faster and more reliable

8.2 TRAINING

Training is only required for the admin who manages this software as to add different brands, he/she needs to edit the master page sections.

Hence a training of 3 months has to be taken by the administrator.

8.3 CONVERSION METHODS

The most secure method for conversion from the old system to the new system is to run the old and new system in parallel. In this approach, a person may operate in the manual older processing system as well as start operating the new computerized system. This method offers

high security, because even if there is a flaw in the computerized system, we can depend upon the manual system. But the cost for maintaining two systems in parallel is very high. Implementation Plan, It identifies the personnel responsible for the activities and prepares a time chart for implementing the system. The implementation plan consists of the following steps:

- List all files required for implementation
- Identify all data required to build new files during the implementation
- List all new documents and procedures that go into the new system

The following are the three types of implementation documents.

- Conversion Guide
- User Guide
- Operation Guide

o CONVERSION GUIDE

The conversion guide phase of the implementation, process the tasks that are required to place the system into an operation mode. They amplify the conversion lane that was defined during the internal design phase and defines file conversion, file creation and data entry requirements.

o USER GUIDE

The system application and operation functions describes the overall performance capabilities of the system and define procedures the user must follow to operate the system. In the realm of information system, the content of a user guide must be developed to coincide with a criterion that defines the characteristics of one of the following methods of data processing.

- Off-line Processing
- Direct Access Processing

o OPERATION GUIDE

The function of an operation is to define the control requirements of a system and provide instruction for initializing, running and terminating the system. The items contained in an operation guide may be grouped as follows.

- General Information
- System Overviews
- Run Description

8.4 POST IMPLEMENTATION REVIEW

After the system is implemented and conversion is completed, a review of system is usually conducted by users and analyst this is called post implementation review.

- The most fundamental concern post implementation review is determining the system has met its objective that is analyst want to know if the performance level of the system has improved and if the system is producing the result intended.
- If neither is happening, one may question whether the system can be considered successful.
- By using current system, all the requirements of all users are fulfilled.

SYSTEM MAINTENANCE

9.1 TYPES OF MAINTENANCE

Maintenance is making adaptation of the software for external changes (requirements changes or enhancements) and internal changes (fixing bugs). When changes are made during the maintenance phase all preceding steps of the model must be revisited. There are three types of maintenance:

- Corrective (Fixing bugs/errors).
- Adaptive (Updates due to environment changes).
- Perfective (Enhancements, requirements changes).

SYSTEM EVALUATION

10 . SYSTEM EVALUATION

- This software is the answer to the existing system of Crime Reporting. This software allows the user to report case and view the Look Out Notice and Notice board through the online website.

- In order to implement this software there are some basic requirements and the minimum requirements are listed above in the system requirements section.

- The software is designed with user-friendliness and attractiveness as the base line for user. Hence a lot of HTML container tags have been designed in order to make it appealing to the user without compromising the software principles.

- Post implementation, the software has to be updated in order to stay ahead of all other competition and to serve the user based on the change in technology.

- There will further addition of different categories and other services linked with real estate management.

- The user is given prime importance and we have designed this software with the ease of use and understanding of this software by the user.

CONCLUSION

11 CONCLUSION

The project entitled “Crime Reporting ” has been successfully engineered and the ability of the system to work has verified. The aim of the project which is to eliminate the drawback of the existing system has been achieved. It has been a great pleasure for me to work on this exciting and challenging project.. It also provides knowledge about the latest technology used in developing web enabled application and client server technology that will be great demand in future. This will provide better opportunities and guidance in future in developing projects independently.

The system has been designed and developed flexibly according to the current requirement of the user. As the information required may still increase shortly. Further such development can be attempted. Several areas are to be developed in the future, so the application must be upgraded for the new once required and it is possible to the modification according to the new requirement and specification. When the organization grows and if its plan is on the increase this system can also be future modified. The basis of the project is well analyzed and prepared so that any changes in the future can be updated to the project.

.

APPENDIX

12.1 SAMPLE CODE

```
<!DOCTYPE html>
{%load static%}
<html lang="en">

<head>
<meta charset="utf-8">
<meta content="width=device-width, initial-scale=1.0" name="viewport">

<title>{% block title %}{% endblock %}</title>
<meta content="" name="description">
<meta content="" name="keywords">

<!-- Favicons -->
<link href="{%static './myapp/assets/img/favicon.png'%}" rel="icon">
<link href="{%static './myapp/assets/img/apple-touch-icon.png'%}" rel="apple-touch-icon">

<!-- Google Fonts -->
<link
href="https://fonts.googleapis.com/css?family=Poppins:300,300i,400,400i,600,600i,700,700i|Sat
isfy|Comic+Neue:300,300i,400,400i,700,700i" rel="stylesheet">

<!-- Vendor CSS Files -->
<link href="{%static './myapp/assets/vendor/animate.css/animate.min.css'%}" rel="stylesheet">
<link href="{%static './myapp/assets/vendor/bootstrap/css/bootstrap.min.css'%}"
rel="stylesheet">
<link href="{%static './myapp/assets/vendor/bootstrap-icons/bootstrap-icons.css'%}"
rel="stylesheet">
<link href="{%static './myapp/assets/vendor/boxicons/css/boxicons.min.css'%}"
rel="stylesheet">
<link href="{%static './myapp/assets/vendor/glightbox/css/glightbox.min.css'%}"
rel="stylesheet">
<link href="{%static './myapp/assets/vendor/swiper/swiper-bundle.min.css'%}"
rel="stylesheet">
```

```
<!-- Template Main CSS File -->
<link href="{%static './myapp/assets/css/style.css'%}" rel="stylesheet">

<!-- =======================================================
* Template Name: Delicious - v4.7.1
* Template URL: https://bootstrapmade.com/delicious-free-restaurant-bootstrap-theme/
* Author: BootstrapMade.com
* License: https://bootstrapmade.com/license/
======================================================= -->
</head>

<body style="background: url({%static './myapp/assets/img/slide/e2.jpg'%}); background-size:
cover">



<!-- ======= Header ======= -->
<header id="header" class="fixed-top d-flex align-items-center header-transparent">
<div class="container-fluid container-xl d-flex align-items-center justify-content-between">

<div class="logo me-auto">
<h1><a href="index.html">Crime Reporting</a></h1>
<!-- Uncomment below if you prefer to use an image logo -->
<!-- <a href="index.html"><img src="assets/img/logo.png" alt="" class="img-fluid"></a>-->
</div>

<nav id="navbar" class="navbar order-last order-lg-0">
<ul>
<li><a class="nav-link scrollto active" href="{% url 'admin_home'%}"
style="color:white">Home</a></li>
<li><a class="nav-link scrollto" href="{%url 'admin_station_master_view'%}"
style="color:white">Station</a></li>
<li class="dropdown"><a href="#" style="color:white"><span>Crime Details</span> <i
class="bi bi-chevron-down"></i></a>
<ul>
```

```
<li><a class="dropdown-item" href="{% url 'admin_state_master_view'%}"
style="color:black">State</a></li>
<li><a class="dropdown-item" href="{% url 'admin_district_master_view2'%}"
style="color:black">District</a></li>
<li><a class="dropdown-item" href="{% url 'admin_place_master_view2'%}"
style="color:black">Place</a></li>
<li><a class="dropdown-item" href="{% url 'admin_crime_type_view'%}"
style="color:black">Crime type</a></li>
</ul>
</li>
<li class="dropdown"><a href="#" style="color:orange"><span>Setting</span> <i class="bi bi-
chevron-down"></i></a>
<ul>
<li><a class="dropdown-item" href="{% url 'admin_changepassword'%}"
style="color:black">Change Password</a></li>
<li><a class="dropdown-item" href="{% url 'admin_logout'%}"
style="color:black">Logout</a></li>
</ul>
</li>
</ul>
<i class="bi bi-list mobile-nav-toggle"></i>
</nav><!-- .navbar -->

</div>
</header><!-- End Header -->
<br>
<br>
<br>
<br><br>
<br>
<br><br>
<br>
<br>
{% block body %}{% endblock %}
<br>
<br>
```

```
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>


<!-- ======= Footer ======= -->
<footer id="footer">
<div class="container" align="center">
<h3 style="color:white">Crime Reporting</h3>
<p style="color:white">Et aut eum quis fuga eos sunt ipsa nihil. Labore corporis magni eligendi fuga maxime saepe commodi placeat.</p>
<div class="social-links">
<a href="#" class="twitter"><i class="bx bxl-twitter"></i></a>
<a href="#" class="facebook"><i class="bx bxl-facebook"></i></a>
<a href="#" class="instagram"><i class="bx bxl-instagram"></i></a>
<a href="#" class="google-plus"><i class="bx bxl-skype"></i></a>
<a href="#" class="linkedin"><i class="bx bxl-linkedin"></i></a>
</div>
</div>



<!DOCTYPE html>
{%load static%}
<html lang="en">

<head>
<style>
label{
```

```
color: White;
}

</style>
<meta charset="utf-8">
<meta content="width=device-width, initial-scale=1.0" name="viewport">

<title>{% block title %}{% endblock %}</title>
<meta content="" name="description">
<meta content="" name="keywords">

<!-- Favicons -->
<link href="{%static './myapp/assets/img/favicon.png'%}" rel="icon">
<link href="{%static './myapp/assets/img/apple-touch-icon.png'%}" rel="apple-touch-icon">

<!-- Google Fonts -->
<link
href="https://fonts.googleapis.com/css?family=Poppins:300,300i,400,400i,600,600i,700,700i|Sat
isfy|Comic+Neue:300,300i,400,400i,700,700i" rel="stylesheet">

<!-- Vendor CSS Files -->
<link href="{%static './myapp/assets/vendor/animate.css/animate.min.css'%}" rel="stylesheet">
<link href="{%static './myapp/assets/vendor/bootstrap/css/bootstrap.min.css'%}"
rel="stylesheet">
<link href="{%static './myapp/assets/vendor/bootstrap-icons/bootstrap-icons.css'%}"
rel="stylesheet">
<link href="{%static './myapp/assets/vendor/boxicons/css/boxicons.min.css'%}"
rel="stylesheet">
<link href="{%static './myapp/assets/vendor/glightbox/css/glightbox.min.css'%}"
rel="stylesheet">
<link href="{%static './myapp/assets/vendor/swiper/swiper-bundle.min.css'%}"
rel="stylesheet">

<!-- Template Main CSS File -->
<link href="{%static './myapp/assets/css/style.css'%}" rel="stylesheet">
```

```
<!-- ======================================================
* Template Name: Delicious - v4.7.1
* Template URL: https://bootstrapmade.com/delicious-free-restaurant-bootstrap-theme/
* Author: BootstrapMade.com
* License: https://bootstrapmade.com/license/
====================================================== -->
</head>

<body style="background: url({%static './myapp/assets/img/slide/e2.jpg'%}); background-size:
cover">



<!-- ======= Header ======= -->
<header id="header" class="fixed-top d-flex align-items-center header-transparent">
<div class="container-fluid container-xl d-flex align-items-center justify-content-between">

<div class="logo me-auto">
<h1><a href="index.html">Crime Reporting</a></h1>
<!-- Uncomment below if you prefer to use an image logo -->
<!-- <a href="index.html"><img src="assets/img/logo.png" alt="" class="img-fluid"></a>-->
</div>

<nav id="navbar" class="navbar order-last order-lg-0">
<ul>
<li><a class="nav-link scrollto active" href="{% url 'index'%}"
style="color:white">Home</a></li>
<li><a class="nav-link scrollto" href="{% url 'index'%}" style="color:white">About</a></li>
<li><a class="nav-link scrollto" href="{% url 'index'%}" style="color:white">Contact</a></li>
<li class="dropdown"><a href="#" style="color:white"><span>Login</span> <i class="bi bi-
chevron-down"></i></a>
<ul>
<li><a class="dropdown-item" href="{% url 'admin_login'%}"
style="color:black">Admin</a></li>
<li><a class="dropdown-item" href="{%url 'station_login'%}"
style="color:black">Police</a></li>
```

```
<li><a class="dropdown-item" href="{% url 'user_login'%}" style="color:black">User</a></li>
</ul>
</li>
</ul>
<i class="bi bi-list mobile-nav-toggle"></i>
</nav><!-- .navbar -->

</div>
</header><!-- End Header -->
<br>
<br>
<br>
<br><br>
<br>
<br><br>
<br>
<br>
{% block body %}{% endblock %}
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<br>



<!-- ======= Footer ======= -->
<footer id="footer">
<div class="container" align="center">
<h3 style="color:white">Crime Reporting</h3>
```

```
<p style="color:white">Et aut eum quis fuga eos sunt ipsa nihil. Labore corporis magni eligendi fuga maxime saepe commodi placeat.</p>
<div class="social-links">
<a href="#" class="twitter"><i class="bx bxl-twitter"></i></a>
<a href="#" class="facebook"><i class="bx bxl-facebook"></i></a>
<a href="#" class="instagram"><i class="bx bxl-instagram"></i></a>
<a href="#" class="google-plus"><i class="bx bxl-skype"></i></a>
<a href="#" class="linkedin"><i class="bx bxl-linkedin"></i></a>
</div>
</div>
</footer><!-- End Footer -->

<a href="#" class="back-to-top d-flex align-items-center justify-content-center"><i class="bi bi-arrow-up-short"></i></a>

<!-- Vendor JS Files -->
<script src="{%static './myapp/assets/vendor/bootstrap/js/bootstrap.bundle.min.js'%}"></script>
<script src="{%static './myapp/assets/vendor/glightbox/js/glightbox.min.js'%}"></script>
<script src="{%static './myapp/assets/vendor/isotope-layout/isotope.pkgd.min.js'%}"></script>
<script src="{%static './myapp/assets/vendor/swiper/swiper-bundle.min.js'%}"></script>
<script src="{%static './myapp/assets/vendor/php-email-form/validate.js'%}"></script>

<!-- Template Main JS File -->
<script src="{%static './myapp/assets/js/main.js'%}"></script>

</body>

</html>
```

BIBLIOGRAPHY

- The complete reference Python Guido van Rossum, and Fred L. Drake.
- SQL – The programming language of oracle ,Ivan Bayross , First edition Developing personal oracle 8i,David Lockmann , second edition
- https://www.freeprojectz.com/python-django-project/auto-spare-parts-management-system
- https://nevonprojects.com/bikers-portal/