

CRIME MANAGEMENT SYSTEM

A

Project Report

Submitted by

Jinto Kurian(Reg No: 200021091400)

Joel George(Reg No: 200021091403)

Balu Anilkumar(Reg No: 200021091387)

In partial fulfillment of the requirements

for

the award of the degree of

BACHELOR OF COMPUTER APPLICATIONS



DEPARTMENT OF COMPUTER APPLICATIONS

YELDO MAR BASELIOS COLLEGE

KOTHAMANGALAM

NOVEMBER 2022

YELDO MAR BASELIOS COLLEGE

(Affiliated to Mahatma Gandhi University)

KOTHAMANGALAM



DEPARTMENT OF COMPUTER APPLICATIONS

Certificate

*This is to certify that the project entitled “**CRIME MANAGEMENT SYSTEM**” submitted in partial fulfillment for the award of the degree of **BACHELOR OF COMPUTER APPLICATION** is a report of the project done by **Balu Anilkumar**, (RegNo.2000) during the year 2022.*

Internal Guide

Head of the department

Asst Prof.

Asst Prof.

Date:

Examiner

1.

2.

Seal

ACKNOWLEDGEMENT

I express my gratitude to **Chev. Prof. Baby M. Varghese**, Principal of Yeldo Mar Baselios College, for providing me with adequate facilities, ways and means by which i am able to complete the project work.

With immense pleasure i take this opportunity to record out sincere thanks to **Asst. Prof. Neethu M Mathew** , Head of the Department, Department of Computer application ,Yeldo Mar Baselios College Kothamangalam.

I especially thank guide Asst. Prof. , Department of Computer Science in examining the draft of this project and suggestions, modifications for having shown keen interest in the development of this project.

I also express my gratitude to all other members of the faculties and well-wishers who assisted me in various occasions during the project work.

Last but not the least, I would like to thank my parents and friends who encouraged me and gave the motivation to complete the project.

Above all my utmost gratitude *The Almighty God* for being with us always.

Balu

TABLE OF CONTENTS

Abstract.....	5
1.Introduction... ..	6
1.1 Introduction... ..	7
1.2 Scope and Relevance of the Project... ..	8
1.3. Objectives... ..	8
2. System Analysis	9
2.1 Introduction to System Analysis... ..	10
2.2 Existing System... ..	10
2.2.1 Limitations of Existing System	10
2.3. Proposed System... ..	11
2.3.1 Advantages of the Proposed System.....	11
2.4 Feasibility Study	12
2.4.1 Technical Feasibility.....	13
2.4.2 Operational Feasibility	13
2.4.3 Economic Feasibility	13
3. System Design	14
3.1 Introduction to System Design.....	15
3.2. Database Design	15
3.2.1 Entity-Relationship Model	22
3.2.2 Data Dictionary	24
3.3 Object-Oriented Design – UML Diagrams	25
3.4 Input Design.....	32
3.5 Output Design.....	39
4. System Environment.....	42
4.1 Introduction System Environment	44
4.2 Software Requirements Specification	46
4.3 Hardware Requirements Specification	46
4.4 Tools, Platforms.....	46
4.4.1 Front End Tool	46
4.4.2 Back End Tool	48
4.4.3 Operating System	49
5. System Implementation	51
5.1 Introduction System Implementation	52
5.2 Coding	54
5.2.1 Coding Standards	54
5.2.2 Sample Codes.....	57
5.3. Debugging	66
5.4 Unit Testing.....	66
6. System Planning and Scheduling	68
6.1 Introduction System Planning and Scheduling.....	69
6.1.1 Planning a Software Project... ..	69
6.1.1.1 Steps Involved in Planning a System... ..	70

7. System Cost Estimation.....	72
7.1 Introduction... ..	73
7.2 LOC Based Estimation/Function Point Based Estimation... ..	74
8. System Maintenance	75
8.1 Introduction System Maintenance	76
8.2 Maintenance.....	76
9. Future Enhancement and Scope of further Development.....	77
9.1 Merits of the System	78
9.2 Limitations of the System	78
9.3 Future Enhancement of the System.....	79

ABSTRACT

This project “Crime Management System” is used to report crimes through mobile application without physically visiting the police station. The aim of this project is to develop an online crime report and managing system which is easily accessible to the public, The police department and the administrative department. The Crime Management System Reporting application is an application that covers complete case management system. This system registers the complaint from people through online and it will also helpful to police department in catching criminals, in system and person can give any complain that any time. It helps to the people who hesitate to visit the police station and let them register complaint online, in this pandemic situation no one is willing to visit the police station and file a complaint so this system will help all those people.

INTRODUCTION

1.1 INTRODUCTION

Crime Management System is an application that helps to reports crime through mobile application. The aim of this project is to develop an Crime Management System and managementsystem which is easily accessible to the public, the police department and the administrative department. The normal public in Kerala is afraid to lodge a complaint because they are filled with a false fear about the police department.

An Crime Management System system will allay the fears of the public and will also help in thepublic helping the police department in catching criminals. The other features of this online solution are enquiry about a complaint and other information. On the whole, the online crime registering and maintenance solution is an excellent method, which bridges the gap between the public and the police department and also helps the police department in preventing criminal activities.

Now days it is time consuming to report a crime. Crime Management System system. We can register a crime without time consuming. It also tracks police station officers' details, police station information, stationratings etc.

1.2. SCOPE AND RELEVANCE OF THE PROJECT

The public were able to know about the updates from the police department and also able to know the complaint status that we are created. So, this system saves time. It provides a facility to generate the reports very easily.

- It satisfies the user requirement
- Be easy to understand by the user and operator
- Be easy to operate
- Have a good user interface
- Be expandable

1.3 OBJECTIVES

Crime Management System system automates all the working process of existing system. This system helps to reduce cost and save time. The public can register into the site without anycost and they can register crimes through online. So, this system saves time. It provides a facility to generate the reports very easily. The public can register complaints by searching with the location. The admin will add the police station details and public will register to the site, later the police station and the public can update their profile very easily. The public can also cancel their complaints and Admin can manage the police station details. The admin can delete the users account if found to be fake or caught for some malpractices.

The Crime Management System system is an application and helps the people to register all type of complaints in a single application with many features and easy to use and simple with the non-complex approach. It helps to the people who hesitate to visit the police station and let them register complaint online, in this pandemic situation no one is willing to visit the police station and file a complaint so this project will help all those people who are quarantined or scared to visit the police station and help them to register any kind of complaints in a single system, this project helps to take a big step in controlling crimes and making corruption less society.

SYSTEM ANALYSIS

2.1 INTRODUCTION TO SYSTEM ANALYSIS

Systems analysis is the process by which an individual (s) studies a system such that an information system can be analysed, modelled, and a logical alternative can be chosen. Systems analysis projects are initiated for three reasons: problems, opportunities, and directives. The people involved include systems analysts, sponsors, and users. The process by which systems are developed can be described by the systems development life cycle. The tasks, techniques, and tools used by the systems development life cycle can be referred as a methodology. There are three classifications of the methodologies: traditional, information engineering, and object-oriented. CASE tools are automated tools that support specific methodologies.

2.2 EXISTING SYSTEM

Currently there is no online application available to report crime online. In order to report any complaints related with crime, people have to contact nearest police station. People of the particular city aren't aware of crime related things such as list of most wanted criminals of their city, latest crime related news, missing person of their area etc. people have to view news channels or read newspapers for such crime related information's. Thus, we can say that existing system is manual and does not provide all the information from one source.

2.2.1 LIMITATIONS OF EXISTING SYSTEM

- Time consuming
- Expensive
- Currently there is no online application for report crime
- Complaint handling is very difficult
- Man work is required

2.3 PROPOSED SYSTEM

- The proposed system overcomes all the limitations of the existing system.
- The administrator of the system can register police station
- Online crime reporting facility
- Information of most wanted criminals
- Information of missing persons
- Police station details and officer details
- Latest crime related news and updates
- Ratings and feedbacks to police station
- Customers can view all details about police station, officers, ratings and feedback

Thus, we can say that new system is fully dynamic and, time saving and requires less man power as compared to existing system.

2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM

- The user interface of this app is user-friendly
- functionality is understandable to the users
- This app gives a complete picture of crime management
- Provide task to improve their response ability
- Cost effective
- Time saving

2.4 FEASIBILITY STUDY

A feasibility study is carried out to select the best system that meets performance requirements. The main aim of the feasibility study activity is to determine whether it would be financially and technically feasible to develop the product. The feasibility study activity involves the analysis of the problem and collection of all relevant information relating to the product such as the different data items which would be input to the system, the processing required to be carried out on these data, the output data required to be produced by the system as well as various constraints on the behaviour of the system. Feasibility analysis is the procedure for identifying candidate system, evaluation and electing the most feasible system. It is a test of a system proposal according to its workability, impact on the organization, ability to meet user needs and effective use of resources. The objective of a feasibility study is not to solve the problem but to acquire a sense of its scope. Feasibility analysis involves 8 steps

- 1) Form a project team and appoint a project leader.
- 2) Prepare system flowcharts.
- 3) Enumerate potential candidate system.
- 4) Describe and identifying characteristics of candidate systems.
- 5) Determination and evaluate performance and cost effectiveness of each candidate system.
- 6) Weight system performance and cost data.
- 7) Select the best candidate system.
- 8) Prepare and report final project directive to management.

2.4.1 TECHNICAL FEASIBILITY

The technical feasibility study is a study of function, performances and constraints and improve the ability to create an acceptable system. Technical feasibility is frequently the most difficult are to achieve at the stage of product engineering process. The system must be evaluated from technical viewpoint first. The assessment of this feasibility must be based on the outline design of the system requirements in the terms of inputs, outputs program procedure and staffs. This project is said to be technically feasible. Technical feasibility centres on the existing computer systems and extend to which it can support the proposed system. This involves financial consideration to technical enhancements. This site is feasible with all aspects of technical.

2.4.2 OPERATIONAL FEASIBILITY

The purpose of the operational feasibility study is to determine the whether the new system will be used if it is developed and installed. And whether there will be resistance from users that will undermine the possible application benefit. The first challenge was whether the system meets the organizational requirements. This is checked by the system requirement collected from the users and the management and the operational feasibility proved that the system is capable to meet its functional requirements. The developed system is completely driven and user friendly. In tour booking operational feasibility is dependent on human resources available for the project and involves projecting whether the system will be used if it is developed and implemented. Operational feasibility is a measure of how proposed system solves the problem and how it satisfies requirements identified in the requirement analysis phase of system development. This site is feasible with all aspects of operation.

2.4.3 ECONOMICAL FEASIBILITY

Economic analysis is the most frequently used method for evaluating the effectiveness of the proposed system. It evaluates whether the system benefits greater than cost. The proposed tour booking system is an effective one since the benefits of the software outweigh the cost incurred in installing it. It can be developed under optimal expenses with the available hardware and software. This site is economically feasible.

SYSTEM DESIGN

3.1 INTRODUCTION TO SYSTEM DESIGN

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), out puts (destinations), databases (data sores) and procedures (data flows) all in a format that meats the uses 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 fulfil. 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
- Form design
- Architectural design
- System modules

3.2 DATABASE DESIGN

Data Base 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 of 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 criterions. 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 thus increase the performance of the system or application which is directly linked to the database design and also solve the problems of anomalies. There are different forms of normalization, some 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)

The database design of the new system is in Second normal form and every non-key attribute is functionally depends only on the primary key. The master and transaction tables and their structure are shown below.

1) Admin

Field	Datatype	Constraint	Description
admin_id	int	Primary key, Auto increment	Admin Id
admin_username	varchar(50)	Not null	Username
admin_password	varchar(30)	Not null	Password

2) District

Field	Datatype	Constraint	Description
dist_id	int	Primary key, Auto increment	District Id
dist_name	varchar(50)	Not null	District name

3) Designation

Field	Datatype	Constraint	Description
des_id	int	Primary key, Auto	Designation Id
		increment	
des_name	varchar(100)	Not null	Designation

4) Awareness

Field	Datatype	Constraint	Description
awareness_id	int	Primary increment , Key , Auto	Awareness id
awareness_title	varchar(200)	Not null	Awareness title
awareness_description	varchar(500)	Not null	Description
awareness_file	varchar(300)	Null	Awareness file
awareness_date	Date	Not null	Date of awareness

5) Customer Reg

Field	Datatype	Constraint	Description
customer_id	int	Primary key, Auto increment	Customer Id
dist_id	Int	Foreign key	District id
customer_name	varchar(50)	Not null	Name
customer_gender	varchar(20)	Not null	Gender
customer_address	varchar(300)	Not null	Address
customer_email	varchar(100)	Not null	Email
customer_phno	varchar(20)	Not null	Phone Number
customer_username	varchar(50)	Not null	Username
customer_password	varchar(30)	Not null	Password
customer_reg_date	Date	Not null	Registration date

6) Rating

Field	Datatype	Constraint	Description
rating_id	Int	Primary key, Auto increment	Rating id
customer_id	Int	Foreign key	Customer id
station_id	Int	Foreign key	Station Id
rating	varchar(50)	Not null	Rating value
feedback	varchar(100)	Null	Feedback

7) Station Reg

Field	Datatype	Constraint	Description
station_id	Int	Primary key, Auto increment	Station id
dist_id	Int	Foreign key	District id
station_name	varchar(100)	Not null	Station name
station_address	varchar(300)	Not null	Station address
station_email	varchar(100)	Not null	Email
station_phno	varchar(20)	Not null	Phone number
station_website	varchar(50)	Null	Website url
station_username	varchar(50)	Not null	Username
station_password	varchar(30)	Not null	Password
station_reg_date	Date	Not null	Registration date
satus	varchar(20)	Not null	Account status

8) Complaints

Field	Datatype	Constraint	Description
complaints_id	Int	Primary key, Auto increment	Complaint id
dist_id	Int	Foreign key	District id
station_id	Int	Foreign key	Station id
customer_id	Int	Foreign key	Customer id
complaint_title	varchar(100)	Not null	Complaint title
complaint_desc	varchar(600)	Not null	Complaint description
complaint_proof	varchar(300)	Null	Proof
complaint_date	Date	Not null	Date of complaint
complaint_status	varchar(50)	Not null	Complaint status

9) Accused

Field	Datatype	Constraint	Description
accused_id	Int	Primary key, Auto increment	Accused id
complaints_id	Int	Foreign key	Complaint id
accused_name	varchar(50)	Null	Name
accused_address	varchar(300)	Null	Address
accused_email	varchar(100)	Null	Email
accused_phno	varchar(20)	Null	Phone Number

10) Officer Reg

Field	Datatype	Constraint	Description
officer_id	Int	Primary key, Auto increment	Officer id
dist_id	Int	Foreign key	District id
station_id	Int	Foreign key	Station id
des_id	Int	Foreign key	Designation id
officer_name	varchar(50)	Not null	Name
officer_address	varchar(300)	Not null	Address
officer_email	varchar(100)	Not null	Email
officer_phno	varchar(20)	Not null	Phone Number

3.2.1 Entity-Relationship Model

An entity relationship diagram (ERD), also known as an entity relationship model, is a graphical representation that depicts relationships among people, objects, places, concepts or events within an information technology (IT) system. An ERD uses data modelling techniques that can help define business processes and serve as the foundation for a relational database.

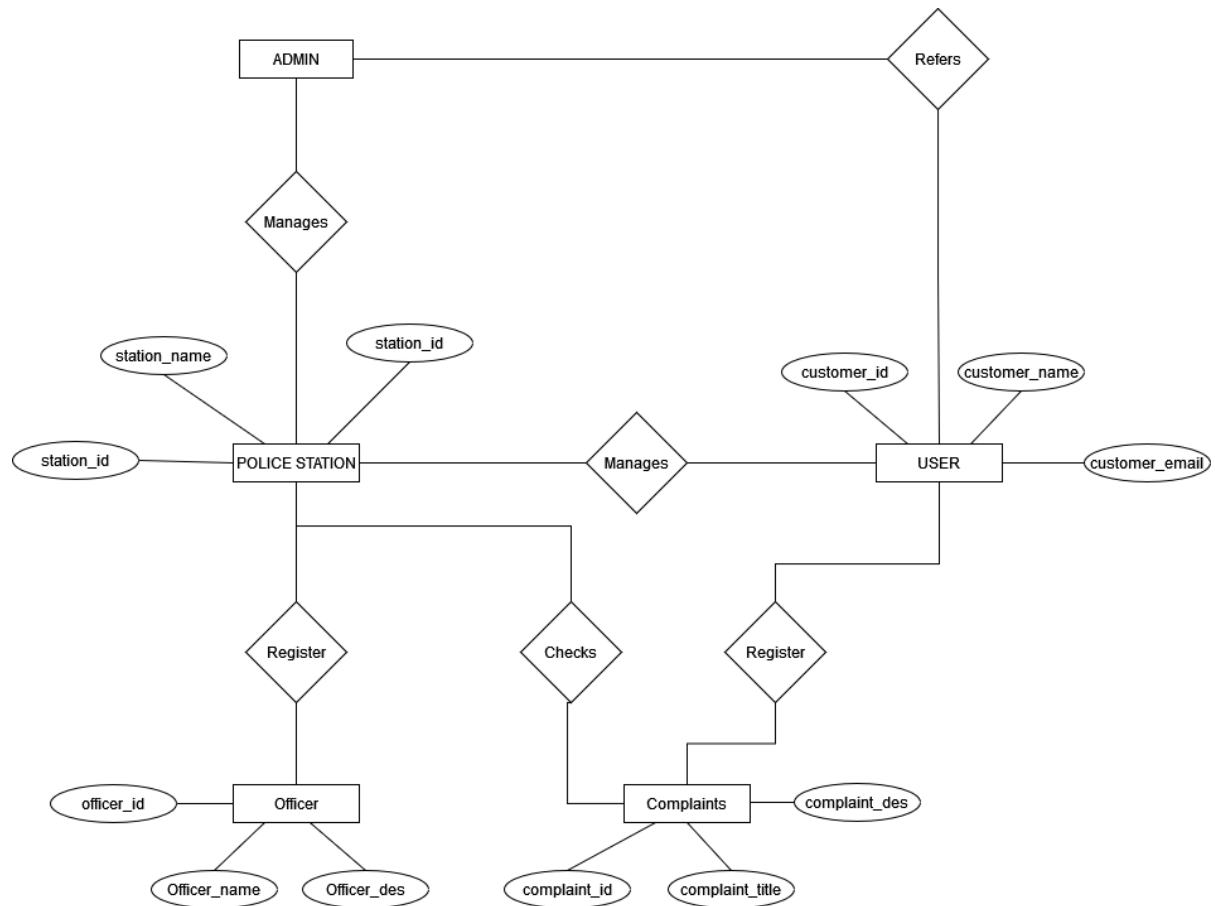
Entity relationship diagrams provide a visual starting point for database design that can also be used to help determine information system requirements throughout an organization. After a relational database is rolled out, an ERD can still serve as a reference point, should any debugging or business process re-engineering be needed later.

However, while an ERD can be useful for organizing data that can be represented by a relational structure, it can't sufficiently represent semi-structured or unstructured data. It's also unlikely to be helpful on its own in integrating data into a pre-existing information system.

ERDs are generally depicted in one or more of the following models:

- A conceptual data model, which lacks specific detail but provides an overview of the scope of the project and how data sets relate to one another.
- A logical data model, which is more detailed than a conceptual data model, illustrating specific attributes and relationships among data points. While a conceptual data model does not need to be designed before a logical data model, a physical data model is based on a logical data model.
- A physical data model, which provides the blueprint for a physical manifestation -- such as a relational database -- of the logical data model. One or more physical data models can be developed based on a logical data model.

ER Diagram for Crime Management System



3.2.1 DATA DICTIONARY

A Data Dictionary is a collection of names, definitions, and attributes about data elements that are being used or captured in a database, information system, or part of a research project. It describes the meanings and purposes of data elements within the context of a project, and provides guidance on interpretation, accepted meanings and representation. A Data Dictionary also provides metadata about data elements. The metadata included in a Data Dictionary can assist in defining the scope and characteristics of data elements, as well the rules for their usage and application.

Data Dictionaries are useful for a number of reasons. In short, they:

- Assist in avoiding data inconsistencies across a project
- Help define conventions that are to be used across a project
- Provide consistency in the collection and use of data across multiple members of a research team
- Make data easier to analyse
- Enforce the use of Data Standards

Data Standards are rules that govern the way data are collected, recorded, and represented. Standards provide a commonly understood reference for the interpretation and use of data sets.

By using standards, researchers in the same disciplines will know that the way their data are being collected and described will be the same across different projects. Using Data Standards as part of a well-crafted Data Dictionary can help increase the usability of your research data, and will ensure that data will be recognizable and usable beyond the immediate research team.

3.3 Object-Oriented Design – UML Diagrams

UML stands for Unified Modelling Language. UML is a language for specifying, visualizing and documenting the system. This is the step while developing any product after analysis. The goal from this is to produce a model of the entities involved in the project which later need to be built. The representation of the entities that are to be used in the product being developed need to be designed. Software design is a process that gradually changes as various new, better and more complete methods with a broader understanding of the whole problem in general come into existence. There are various kinds of methods in software design. They are as follows:

- Use case diagram
- Activity diagram
- Sequence diagram
- Class diagram

Use case Diagrams:

Use case diagrams model behaviour within a system and helps the developers understand of what the user require. The stick man represents what's called an actor. An actor represents an outside entity- either human or technological. Use case diagrams can be useful for getting an overall view of the system and clarifying who can do and more importantly what they can't do. Use case Diagram consists of use cases and actors and shows the interaction between the use case and actors. The purpose is to show the interactions between use cases and actor. To represent the system requirements from user's perspective. It must be remembered that the use-cases are the functions that are to be performed in the module. An actor could be the end-user of the system or an external system.

Activity Diagram:

The purpose is to show the activities which the users performed. Activities are shown parallel and sequentially in which order they are performed. Some activities are joined and split according to its flow. Flow of data is represented using arrows.

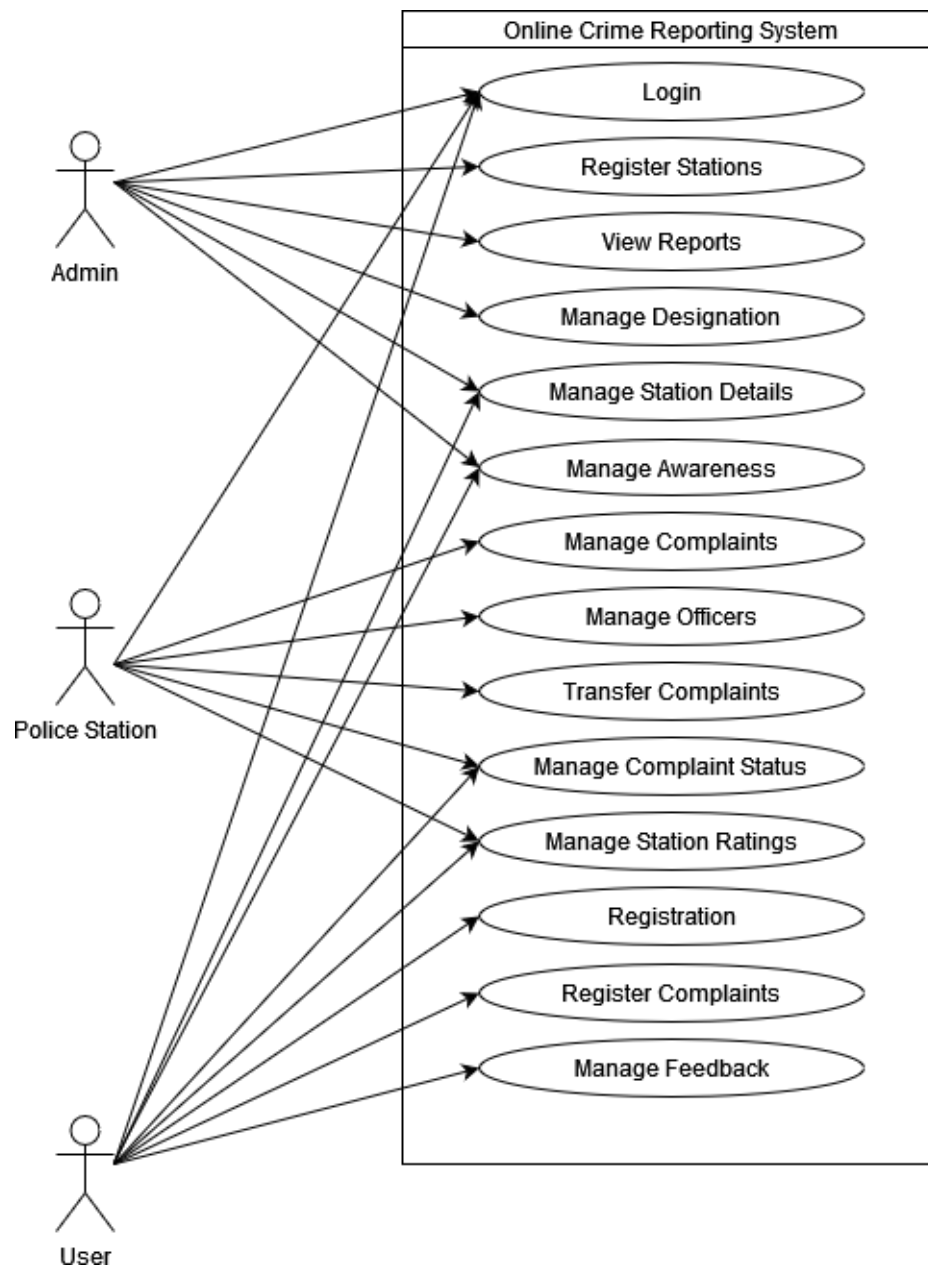
Sequence Diagram:

The purpose is to show the sequential flow through of activities. In other Words, we call it mapping processes in terms of data transfers from the actor through corresponding objects. To represent the logical flow of data with respect to a process. It must be remembered that the sequence diagram display objects and not the classes.

Class Diagram:

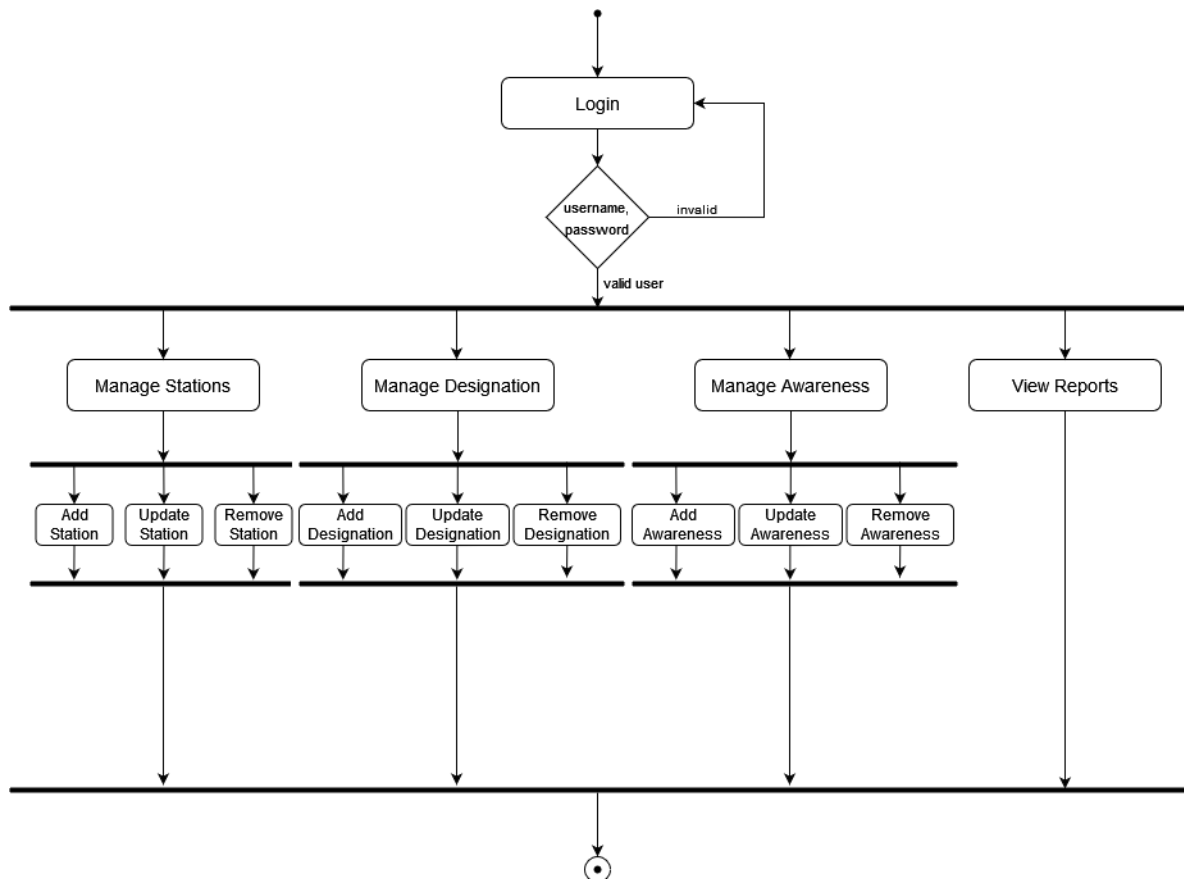
This is one of the most important of the diagrams in development. The diagram breaks the class into three layers. One has the name, the second describes its attributes and the third its methods. The private attributes are represented by a padlock to left of the name. The relationships are drawn between the classes. Developers use the Class Diagram to develop the classes. Analyses use it to show the details of the system. Architects look at class diagrams to see if any class has too many functions and see if they are required to be split.

1) USECASE DIAGRAM

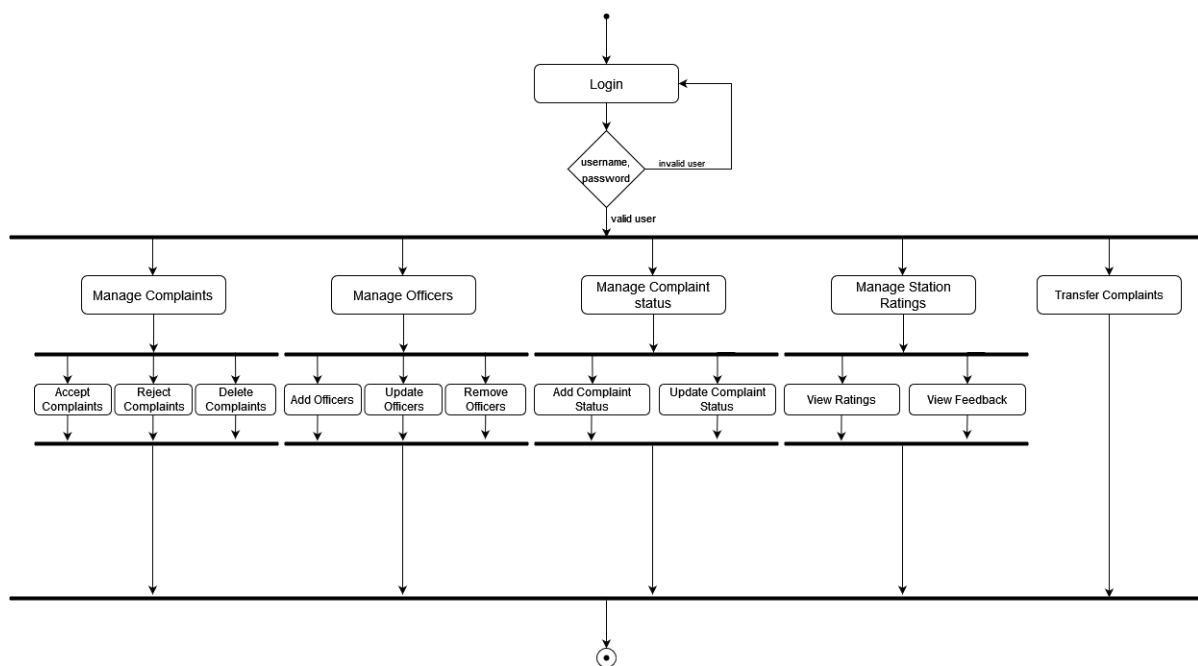


2) ACTIVITY DIAGRAM

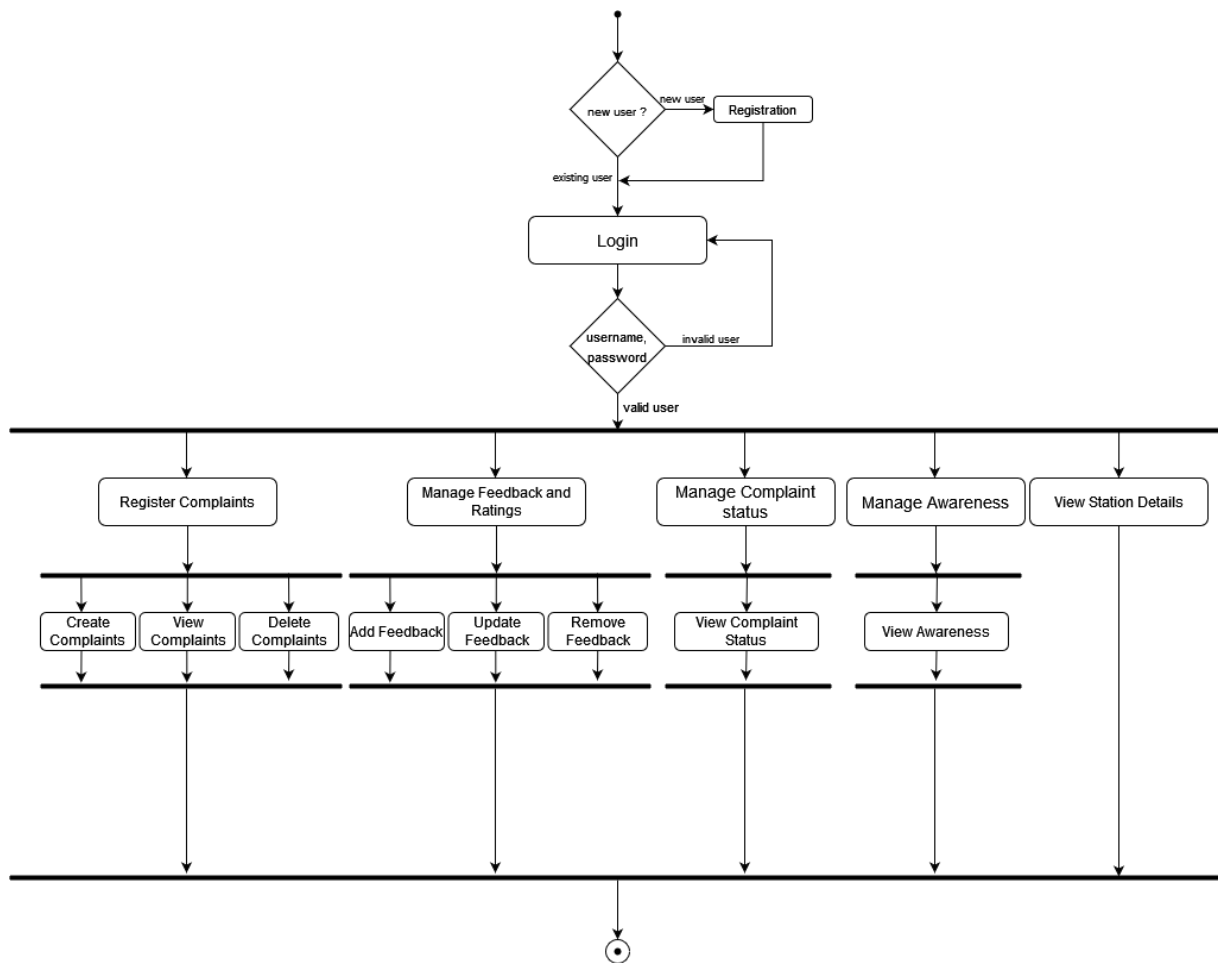
a) ADMIN



b) POLICE STATION



c) USER

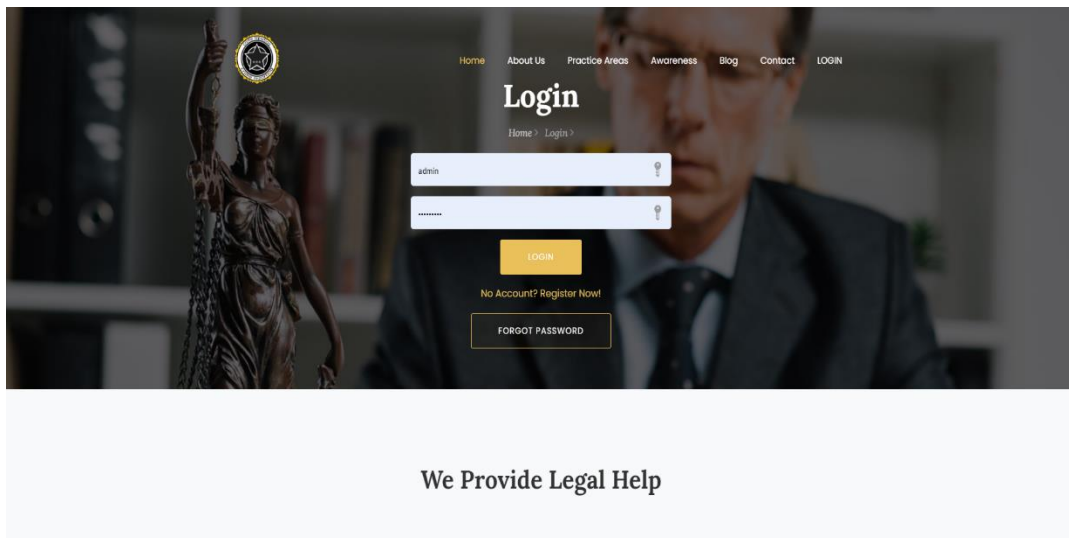


3.4 INPUT DESIGN

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.

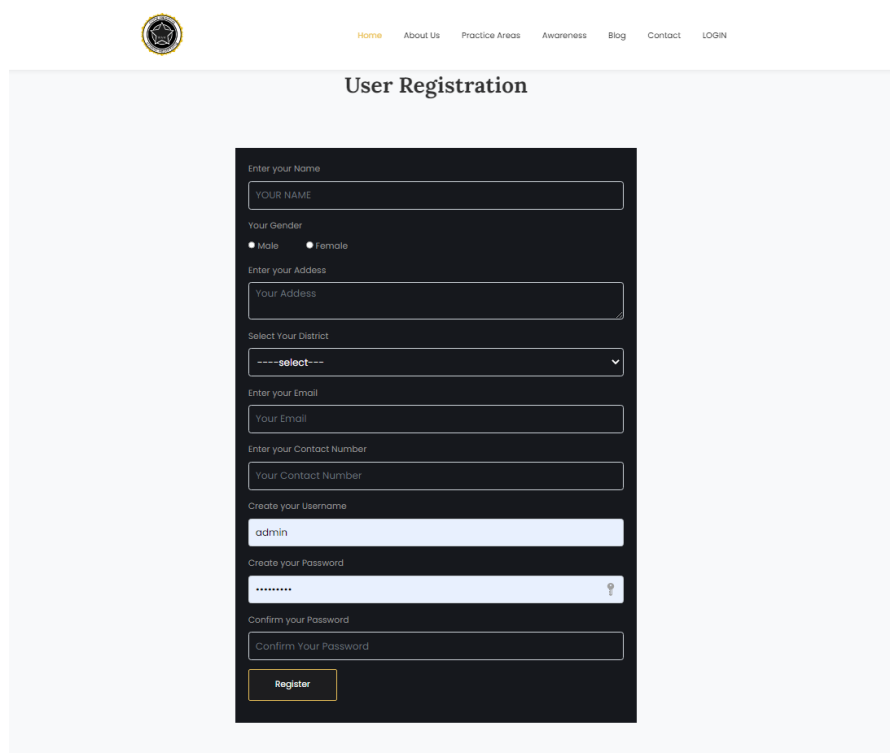
- What data should be given as input?
- The dialogue to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error Occur.
- **Login Form**

The login form contains two text fields, one for the user to enter a username and the other to enter the password. Once the details are entered, the login button is clicked to login. On clicking the cancel button, the fields are cleared.



- **Registration Form**

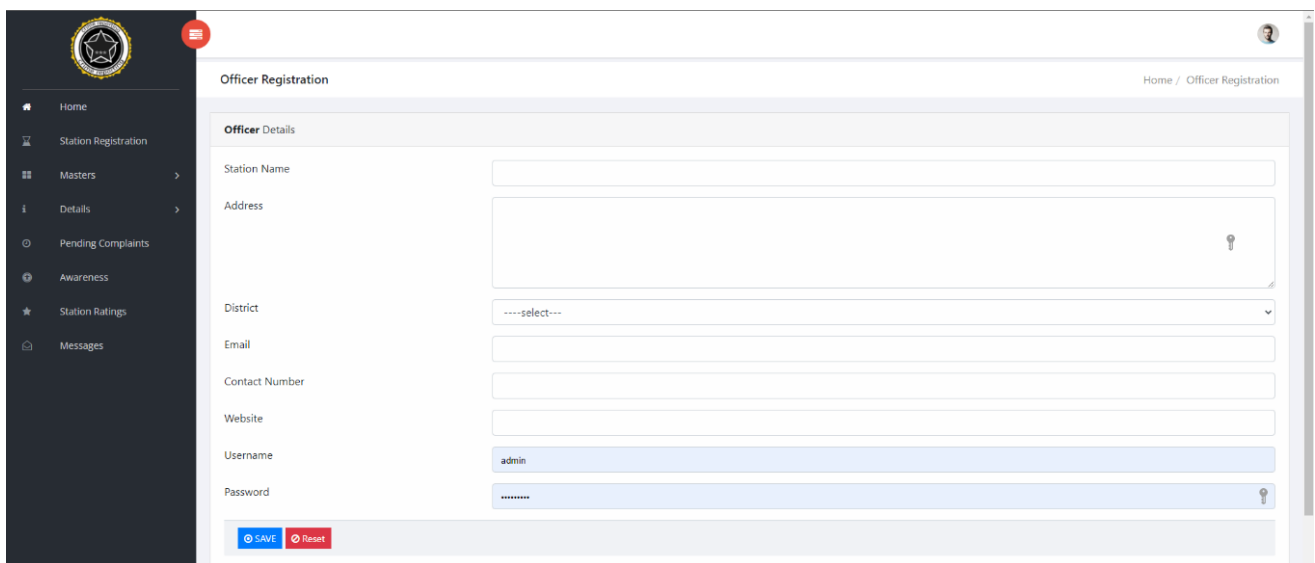
This form used for the registration purpose. User can create account using the registration form.



The image shows a screenshot of a web application's user registration form. At the top left is a circular logo. To its right is a horizontal navigation menu with links: Home, About Us, Practice Areas, Awareness, Blog, Contact, and LOGIN. The main heading of the form is "User Registration". The form itself is a dark-themed overlay with the following fields and elements: "Enter your Name" with a text input containing "YOUR NAME"; "Your Gender" with radio buttons for "Male" and "Female"; "Enter your Address" with a text input containing "Your Address"; "Select Your District" with a dropdown menu showing "----select---"; "Enter your Email" with a text input containing "Your Email"; "Enter your Contact Number" with a text input containing "Your Contact Number"; "Create your Username" with a text input containing "admin"; "Create your Password" with a password input containing "*****" and a toggle icon; "Confirm your Password" with a text input containing "Confirm Your Password"; and a yellow "Register" button at the bottom.

- **Register Stations**

With the use of this form admin can register new police stations to the system. Some fields like station name, address, details, username, password are required to register stations.



The screenshot displays a web application interface for 'Officer Registration'. On the left is a dark sidebar with a logo and a menu containing: Home, Station Registration, Masters, Details, Pending Complaints, Awareness, Station Ratings, and Messages. The main content area is titled 'Officer Registration' and 'Officer Details'. It contains a form with the following fields: Station Name (text input), Address (text input with a key icon), District (dropdown menu showing '---select---'), Email (text input), Contact Number (text input), Website (text input), Username (text input with 'admin' pre-filled), and Password (text input with masked characters and a key icon). At the bottom of the form are two buttons: 'SAVE' (blue) and 'Reset' (red). The top right of the page shows a breadcrumb 'Home / Officer Registration' and a user profile icon.

- **Manage Officers**

This form is used for managing the officers by the police station. Police station can add, view, edit and delete the officers.

The screenshot displays the 'Officer Edit' interface within a police management system. On the left is a dark sidebar with a navigation menu containing 'Home', 'Complaints', 'Officer Registration', 'Details', and 'Ratings'. The main content area is titled 'Officer Edit' and shows the following details:

- Officer Details**
- Name:** BIJU POULOSE
- Address:** Abcd House, Aluva PO, Ernakulam
- Email:** cidbiju@gmail.com
- Phone Number:** 9985467895
- Designation:** POLICE SUB INSPECTOR

At the bottom of the form are two buttons: 'Submit' (blue) and 'Reset' (red). A 'Back' button is located below the form area. The top right corner of the page shows a breadcrumb trail: 'Home / Officer Edit'.

- **Transfer Complaints**

This form is used to transfer a complaint from one station to another.

The screenshot shows a web application interface for transferring a complaint. On the left is a dark sidebar with a logo at the top and a menu with items: Home, Complaints, Officer Registration, Details, and Ratings. The main content area is titled 'Complaint Transfer' and includes a breadcrumb 'Home / Complaint Transfer'. The form itself is titled 'Complaint ID : 1' and contains the following fields: 'Complaint Title' with the value 'TEST', 'From' section with 'Station Name' as 'ALUVA', and 'To' section with 'District' and 'Station' both set to 'WAYANAD' via dropdown menus. At the bottom of the form are 'Transfer' and 'Reset' buttons. A 'Back' button is located below the form. In the bottom right corner of the page, there is a watermark that says 'Activate Windows Go to Settings to activate Windows.'

- **Add Awareness**

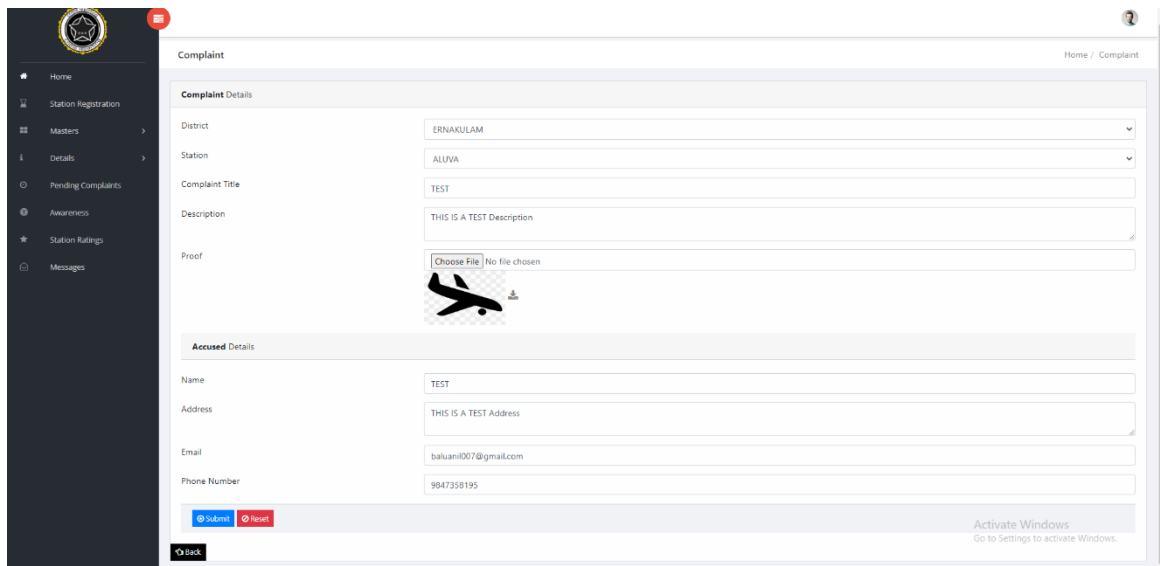
This form is used to add updates, awareness and useful tips to the users

The screenshot displays a web application interface for adding awareness. On the left is a dark sidebar with navigation links: Home, Station Registration, Masters, Details, Pending Complaints, Awareness (selected), Station Ratings, and Messages. The main content area is titled 'Awareness' and contains an 'Awareness Details' form. The form has three input fields: 'Title' with the value 'LINKS', 'Description' with the value 'Dont click unwanted link', and 'File' with a 'Choose File' button and a small image thumbnail. Below the form are 'Submit' and 'Cancel' buttons. Underneath the form is a 'Data Table' with a search bar and a table listing awareness entries. The table has columns for No, Title, Description, File, Date, Edit, and Delete. One entry is visible with No. 3, Title 'LINKS', and Description 'Dont click unwanted link'. The Date is '2023-01-19'. The Edit and Delete columns contain icons for editing and deleting the entry.

No	Title	Description	File	Date	Edit	Delete
3	LINKS	Dont click unwanted link		2023-01-19		

- **Complaints**

This form is used to file a complaint to the nearest police station by searching the location.



The screenshot displays a web application interface for filing a complaint. On the left is a dark sidebar with a police emblem and a menu containing: Home, Station Registration, Masters, Details, Pending Complaints, Awareness, Station Ratings, and Messages. The main content area is titled 'Complaint' and includes a breadcrumb 'Home / Complaint'. It is divided into two sections: 'Complaint Details' and 'Accused Details'. The 'Complaint Details' section contains fields for District (ERNAKULAM), Station (ALUVA), Complaint Title (TEST), Description (THIS IS A TEST Description), and Proof (with a 'Choose File' button and a placeholder image of an airplane). The 'Accused Details' section contains fields for Name (TEST), Address (THIS IS A TEST Address), Email (baluani007@gmail.com), and Phone Number (9847358195). At the bottom of the form are 'Submit' and 'Reset' buttons, and a 'Logout' button in the sidebar. A Windows watermark is visible in the bottom right corner.

Complaint Details	
District	ERNAKULAM
Station	ALUVA
Complaint Title	TEST
Description	THIS IS A TEST Description
Proof	Choose File No file chosen

Accused Details	
Name	TEST
Address	THIS IS A TEST Address
Email	baluani007@gmail.com
Phone Number	9847358195

Submit Reset Logout

Activate Windows
Go to Settings to activate Windows.

3.5 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
- **View Station Details**

It provides the details about police stations that are registered in the site. the admin can also delete the station account if any fraud or malpractice identified.

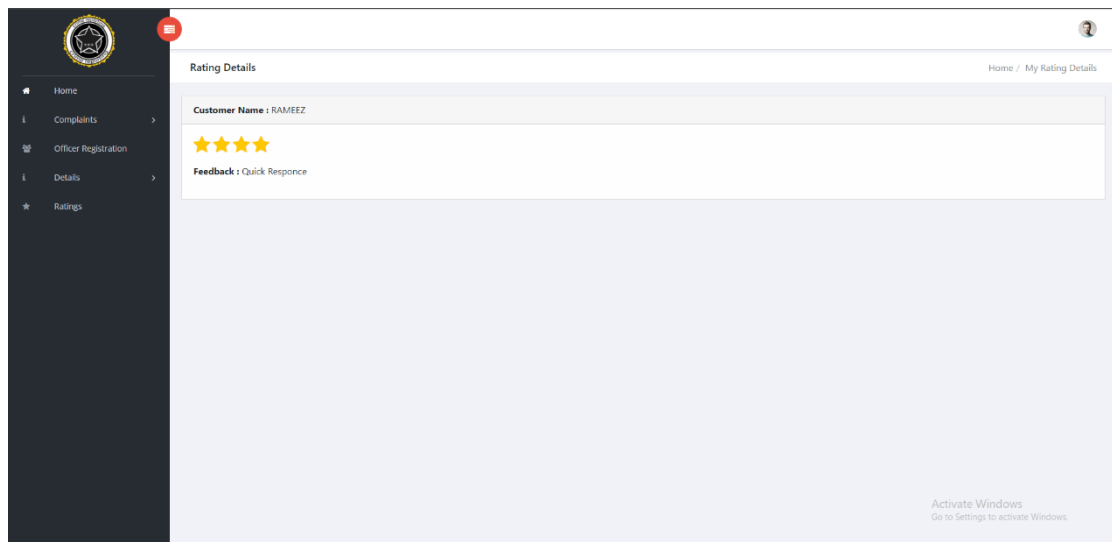
Station

Home / Station

Data Table					
Show 10 entries			Search: <input type="text"/>		
No <small>↑↓</small>	ID <small>↑↓</small>	Station Name <small>↑↓</small>	District <small>↑↓</small>	Registration Date <small>↑↓</small>	View <small>↑↓</small>
1	1	ALUVA	ERNAKULAM	2019-12-15	
2	2	ANGAMALY	ERNAKULAM	2019-12-18	
3	3	ALUVA	ERNAKULAM	2020-05-21	

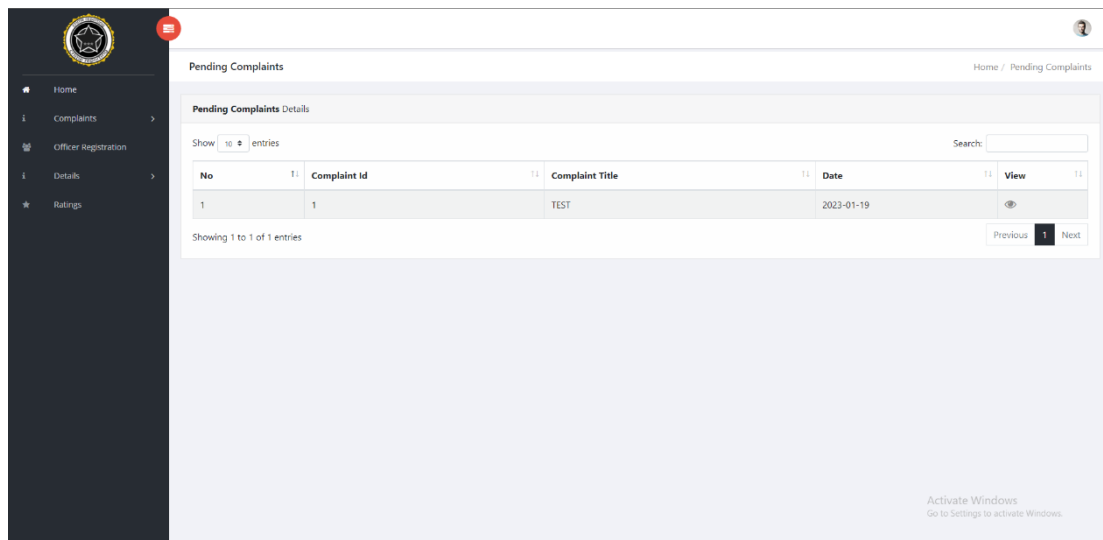
- **View feedbacks**

It provides the details of the feedback. The police stations can view their feedbacks.



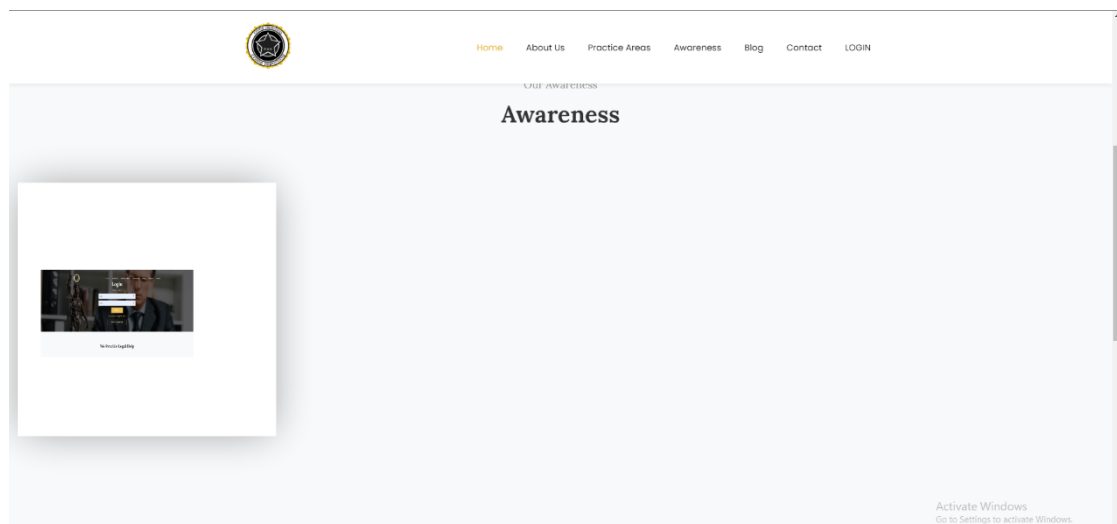
- **View Complaints**

The user and police station can view the complaints. The police station can accept or reject the complaints. The user can cancel the complaint also.



- **View Awareness**

The users are able to view the awareness. The admin can add awareness, useful tips and updates.



SYSTEM ENVIRONMENT

4.1 INTRODUCTION TO SYSTEM ENVIRONMENT

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware. A hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in case of operating systems. An HCL lists tested, compatible, and sometimes incompatible hardware devices for a particular operating system or application. The following sub-sections discuss the various aspects of hardware requirements.

Architecture

All computer operating systems are designed for a particular computer architecture. Most software applications are limited to particular operating systems running on particular architectures. Although architecture-independent operating systems and applications exist, most need to be recompiled to run on a new architecture. See also a list of common operating systems and their supporting architectures.

Processing power

The power of the central processing unit (CPU) is a fundamental system requirement for any software. Most software running on x86 architecture define processing power as the model and the clock speed of the CPU. Many other features of a CPU that influence its speed and power, like bus speed, cache, and MIPS are often ignored. This definition of power is often erroneous, as AMD Athlon and Intel Pentium CPUs at similar clock speed often have different throughput speeds. Intel Pentium CPUs have enjoyed a considerable degree of popularity, and are often mentioned in this category.[citation needed]

Memory

All software, when run, resides in the random access memory (RAM) of a computer. Memory requirements are defined after considering demands of the application, operating system, supporting software and files, and other running processes. Optimal performance of other unrelated software running on a multi-tasking computer system is also considered when defining this requirement.

Secondary storage

Data storage device requirements vary, depending on the size of software installation, temporary files created and maintained while installing or running the software, and possible use of swap space (if RAM is insufficient).

Display adapter

Software requiring a better than average computer graphics display, like graphics editors and high-end games, often define high-end display adapters in the system requirements.

Peripherals

Some software applications need to make extensive and/or special use of some peripherals, demanding the higher performance or functionality of such peripherals. Such peripherals include CD-ROM drives, keyboards, pointing devices, network devices, etc.

Software requirements

Software requirements deal with defining software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or prerequisites are generally not included in the software installation package and need to be installed separately before the software is installed.

Platform

A computing platform describes some sort of framework, either in hardware or software, which allows software to run. Typical platforms include a computer's architecture, operating system, or programming languages and their runtime libraries.

Operating system is one of the requirements mentioned when defining system requirements (software). Software may not be compatible with different versions of same line of operating systems, although some measure of backward compatibility is often maintained. For example, most software designed for Microsoft Windows XP does not run on Microsoft Windows 98, although the converse is not always true. Similarly, software designed using newer features of Linux Kernel v2.6 generally does not run or compile properly (or at all) on Linux distributions using Kernel v2.2 or v2.4.

APIs and drivers

Software making extensive use of special hardware devices, like high-end display adapters, needs special API or newer device drivers. A good example is DirectX, which is a collection of APIs for handling tasks related to multimedia, especially game programming, on Microsoft platforms.

Web browser

Most web applications and software depend heavily on web technologies to make use of the default browser installed on the system. Microsoft Internet Explorer is a frequent choice

of software running on Microsoft Windows, which makes use of ActiveX controls, despite their vulnerabilities.

4.2 RECOMMENDED SOFTWARE REQUIREMENTS SPECIFICATION

Operating System: Windows 10

Database: MySQL

Software used: Visual Studio Code (VS Code)

Front END: HTML, CSS, JAVASCRIPT

Backend: PHP

4.3 RECOMMENDED HARDWARE REQUIREMENTS SPECIFICATION

Processor: Intel® Core (TM) i5-8300H CPU @ 2.30GHz

RAM :8 GB RAM

MONITOR: Standard colour monitor

Keyboard: Standard Keyboard.

Mouse: Standard two button or higher.

4.4 TOOLS, PLATFORMS

4.4.1 FRONT END TOOL

The front-end of an application is distinctly human. It's what the user sees, touches and experiences. In this respect, empathy is a required characteristic of a good front-end developer. The front-end of an application is less about code and more about how a user will interpret the interface into an experience. That experience can be the difference between a billion-dollar company and complete collapse. If you were a Myspace user in 2004, you were probably content with the experience. But once you started to use Facebook, you almost certainly had a better experience. You realized that you could socialize with a simpler design, no flashing banner ads, easy-to-find friends, etc. Facebook and Myspace

had a lot of differences under the hood as well (back-end), but at least part of Facebook's triumph can be attributed to a better front-end and user experience.

The technologies used in front-end development commonly include:

HTML – All code in a web application is eventually translated to HTML. It's the language that web browsers understand and use to display information to users. A web developer's understanding of HTML is analogous to a carpenter's understanding of a screwdriver. It's so important and necessary that it's often assumed for employment.

CSS – By itself, HTML is quite plain. HTML does provide some basic style options, but to build a good front-end, developers must have experience with CSS. CSS provides the paint, templates, glitter, buttons, tassel, lights, and many other things that can be used to improve the presentation of a web page.

CSS is so commonly used that languages have been built to make writing CSS easier. These languages – like Sass and LESS – are also known as CSS pre-compilers, but they are simply used to write more efficient and manageable CSS code.

JavaScript – If you could only learn one language in your lifetime, you'd be well-advised to choose JavaScript. Though it's not exclusively a front-end language, that's where it's most commonly used. JavaScript is a language that is run on a client machine, i.e. a user's computer. This means that JavaScript can be used to program fast, intuitive and fun user experiences, without forcing a user to refresh their web page. Drag-and-drop, infinite-scroll and videos that come to life on a web page can all be programmed with JavaScript. JavaScript is so popular that entire frameworks have been built just to make building application front-ends easier. Frameworks like Angular, Ember, React and Backbone are all very widely used for JavaScript-heavy front-ends.

4.4.2 BACK-END TOOL

The back-end of a web application is an enabler for a front-end experience. An application's front-end may be the most beautifully crafted web page, but if the application itself doesn't work, the application will be a failure. The back-end of an application is responsible for things like calculations, business logic, database interactions, and performance. Most of the code that is required to make an application work will be done on the back-end. Back-end code is run on the server, as opposed to the client. This means that back-end developers not only need to understand programming languages and databases, but they must have an understanding of server architecture as well. If an application is slow, crashes often, or constantly throw errors at users, it's likely because of back-end problems.

PHP- PHP is an open-source server-side scripting language that many devs use for web development. It is also a general-purpose language that you can use to make lots of projects, including Graphical User Interfaces (GUIs). The abbreviation PHP initially stood for

Personal Homepage. But now it is a recursive acronym for Hypertext Pre-processor. (It's recursive in the sense that the first word itself is an abbreviation, so the full meaning doesn't follow the abbreviation.) The first version of PHP was launched 26 years ago. Now it's on version 8, released in November 2020, but version 7 remains the most widely used. PHP runs on the Zend engine, which is the most popular implementation. There are some other implementations as well, like parrot, HPVM (Hip Hop Virtual Machine), and Hip Hop, created by Facebook. PHP is mostly used for making web servers. It runs on the browser and is also capable of running in the command line. So, if you don't feel like showing your code output in the browser, you can show it in the terminal. PHP is platform-independent. You don't have to have a particular OS to use it because it runs on every platform, whether it's Mac, Windows, or Linux. PHP is open source. The original code is made available to everyone who wants to build upon it. This is one of the reasons why one of its frameworks, Laravel, is so popular.

4.4.3 OPERATING SYSTEM

WINDOWS 10

Windows 10 is a series of personal computer operating systems produced by Microsoft as part of its Windows NT family of operating systems. It is the successor to Windows 8.1, and was released to manufacturing on July 15, 2015, and broadly released for retail sale 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 lifespan of extended support.

One of Windows 10's most notable features is support for universal apps, an expansion of the Metro-style apps first introduced in Windows 8. Universal apps can be designed to run across multiple Microsoft product families with nearly identical code—including PCs, tablets, smartphones, embedded systems, Xbox One, Surface Hub and Mixed Reality. 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.

Windows 10 received mostly positive reviews upon its original release in July 2015. Critics praised Microsoft's decision to provide a desktop-oriented interface in line with previous versions of Windows, contrasting the tablet-oriented approach of 8, although Windows 10's touch-oriented user interface mode was criticized for containing regressions upon the touch-oriented interface of Windows 8. Critics also praised the improvements to Windows 10's bundled software over Windows 8.1, Xbox Live integration, as well as the functionality and capabilities of the Cortana personal assistant and the replacement of Internet Explorer with Edge. However, media outlets have been critical of changes to operating system behaviors, including mandatory update installation, privacy concerns over data collection performed by the OS for Microsoft and its partners and the adware-like tactics used to promote the operating system on its release.

SYSTEM IMPLEMENTATION

5.1 INTRODUCTION TO SYSTEM IMPLEMENTATION

A crucial phase in the system 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 before the system is up and running. A crucial factor in conversion is not disrupting the functioning of organization. Actual data were input into the program 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 therefore be carefully planned and controlled to avoid problems. The implementation phase involves the following tasks:

- Careful planning.
- Investigation
- Design of methods
- 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 the ongoing system, but only computerizing the whole system to make the work, evaluation and retrieval of data easier, faster and reliable.

TRAINING

System implementation is the process of making the newly designed system fully operational and consistent in performance. The logical miss-working the system can be identified if any. Various combinations of test data were feed. Each process accuracy/reliability checking was made. After the approval, the system was implemented in the user department.

The preparation of implementation of documentation process is often viewed as total sum of the software documentation process. In a well-defined software development environment, however the presentation of implementation documents is essentially an

interactive process that synthesis and recognizes document items that were produced during the analysis and design phase for the presentation to user.

CONVERSION METHODS

The following are the three types of implementation documents.

- Conversion Guide
- User Guide
- Operation Guide

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.

User Guide

The system application and operation functions describe 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

Operation Guide

The function of an operation is to define the control requirements of an online tour booking website 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

POST IMPLEMENTATION REVIEW

Mental Health Prediction application development has successfully completed. And after completion it runs under several test conditions and checking all the proposed features are completed or not. If functions are missing tried to implement the missing functions to an extent. After the development of application, it runs under a trial run for a certain number of days and this for identifying the crashes and bugs and after the trial run it runs smoothly and some bugs were there and it fixed. And for the notification part admin sent several notifications to the users end and it worked smoothly as well.

5.2 CODING

Coding, sometimes called computer programming, is how we communicate with computers. Code tells a computer what actions to take, and writing code is like creating a set of instructions. By learning to write code, you can tell computers what to do or how to behave in a much faster way. You can use this skill to make websites and apps, process data, and do lots of other cool things.

5.2.1 CODING STANDARDS

Different modules specified in the design document are coded in the Coding phase according to the module specification. The main goal of the coding phase is to code from the design document prepared after the design phase through a high-level language and then to unit test this code.

Good software development organizations want their programmers to maintain to some well-defined and standard style of coding called coding standards. They usually make their own coding standards and guidelines depending on what suits their organization best and based on the types of software they develop. It is very important for the programmers to maintain the coding standards otherwise the code will be rejected during code review.

Purpose of Having Coding Standards:

- A coding standard gives a uniform appearance to the codes written by different engineers.
- It improves readability, and maintainability of the code and it reduces complexity also.

- It helps in code reuse and helps to detect error easily.
- It promotes sound programming practices and increases efficiency of the programmers.

Some of the coding standards are given below:

1. Limited use of globals:

These rules talk about which types of data that can be declared global and the data that can't be.

2. Standard headers for different modules:

For better understanding and maintenance of the code, the header of different modules should follow some standard format and information. The header format must contain below things that is being used in various companies:

- Name of the module
- Date of module creation
- Author of the module
- Modification history
- Synopsis of the module about what the module does
- Different functions supported in the module along with their input output parameters
- Global variables accessed or modified by the module

3. Naming conventions for local variables, global variables, constants and functions:

Some of the naming conventions are given below:

- Meaningful and understandable variables name help anyone to understand the reason of using it.
- Local variables should be named using camel case lettering starting with small letter (e.g., **local Data**) whereas Global variables names should start with a capital letter (e.g., **Global Data**). Constant names should be formed using capital letters only (e.g., **CONSDATA**).
- It is better to avoid the use of digits in variable names.
- The names of the function should be written in camel case starting with small letters.
- The name of the function must describe the reason of using the function clearly and briefly.

4. Indentation:

Proper indentation is very important to increase the readability of the code. For making the code readable, programmers should use White spaces properly. Some of the spacing conventions are given below:

- There must be a space after giving a comma between two function arguments.
- Each nested block should be properly indented and spaced.
- Proper Indentation should be there at the beginning and at the end of each block in the program.
- All braces should start from a new line and the code following the end of braces also start from a new line.

5. Error return values and exception handling conventions:

All functions that encountering an error condition should either return a 0 or 1 for simplifying the debugging.

On the other hand, Coding guidelines give some general suggestions regarding the coding style that to be followed for the betterment of understandability and readability of the code. Some of the coding guidelines are given below:

6. Avoid using a coding style that is too difficult to understand:

Code should be easily understandable. The complex code makes maintenance and debugging difficult and expensive.

7. Avoid using an identifier for multiple purposes:

Each variable should be given a descriptive and meaningful name indicating the reason behind using it. This is not possible if an identifier is used for multiple purposes and thus it can lead to confusion to the reader. Moreover, it leads to more difficulty during future enhancements.

8. Code should be well documented:

The code should be properly commented for understanding easily. Comments regarding the statements increase the understandability of the code.

9. Length of functions should not be very large: Lengthy functions are very difficult to understand. That's why functions should be small enough to carry out small work and lengthy functions should be broken into small ones for completing small tasks.

5.2.2 SAMPLE CODES

Login

```
<?php
require "header.php";

if(isset($_POST['btnsubmit']))
{
    $selA="select * from tbl_admin where admin_username='".$_strtolower($_POST['name'])."'
and admin_password='".$_$_POST['password'].''";
    $rowA=mysqli_query($con,$selA);
    $numA=mysqli_num_rows($rowA);

    $selE="select * from tbl_station_reg where
station_username='".$_strtolower($_POST['name'])."' and station_password='".$_$_POST['password'].''";
    $rowE=mysqli_query($con,$selE);
    $numE=mysqli_num_rows($rowE);

    $selC="select * from tbl_customer_reg where
customer_username='".$_strtolower($_POST['name'])."' and
customer_password='".$_$_POST['password'].''";
    $rowC=mysqli_query($con,$selC);
    $numC=mysqli_num_rows($rowC);

    if($numA>0)
    {
        $dataA=mysqli_fetch_assoc($rowA);
        $_SESSION['LoginId']=1;
        $_SESSION['Id']=$dataA['admin_id'];
        header("Location:Dashboard/Pages/Others/index.php");
    }
    else if($numE>0)
    {
        $dataA=mysqli_fetch_assoc($rowE);
        if($dataA['status']=='1')
        {
            $_SESSION['LoginId']=2;
            $_SESSION['Id']=$dataA['station_reg_id'];
            header("Location:Dashboard/Pages/Others/index.php");
        }
        else
        {
            echo "<script>alert('Accout Is Not Activated...')</script>";
        }
    }
}
```

```

        else if($numC>0)
        {
            $dataA=mysqli_fetch_assoc($rowC);
            $_SESSION['LoginId']=3;
            $_SESSION['Id']=$dataA['customer_id'];
            header("Location:Dashboard/Pages/Others/index.php");
        }
        else{
            echo "<script>alert('Invalid Username or Password')</script>";
        }
    }

    if(isset($_POST['btnforgot']))
    {
        header("Location:Forgot.php");
        //$_SESSION['ForName']=$_POST['name'];
        //header("Location:Email.php");
    }

    if(isset($_POST['btnuser']))
    {
        header("Location:UserRegistration.php");
    }
?>

<script src="jQuery.js" type="text/javascript"></script>
<script>
function getReg()
{
    $.ajax({
        url:"AjaxRegistration.php",
        success:function(result)
        {
            $("#list").html(result);
        }
    });
}
</script>

<section class="hero-wrap hero-wrap-2" style="background-image: url('images/bg_2.jpg');"
data-stellar-background-ratio="0.5">
    <div class="overlay"></div>
    <div class="container">
        <div class="row no-gutters slider-text align-items-end justify-content-center">
            <div class="col-md-5 ftco-animate pb-5 text-center" style="margin-top: 100px;">
                <h1 class="mb-3 bread">Login</h1>
            </div>
        </div>
    </div>

```

```

        <p class="breadcrumbs"><span class="mr-2"><a href="index.php">Home <i
class="ion-ios-arrow-forward"></i></a></span> <span>Login <i class="ion-ios-arrow-
forward"></i></span></p>
        <form action="#" method="post" class="contact-form">
        <div class="form-group">
            <input type="text" name="name" style="color:#fff !important;background-
color:transparent !important" class="form-control" placeholder="Your Username"
minlength="4" maxlength="30" style="text-transform: lowercase;" pattern="[A-Za-z0-
9]{4,30}" title="Enter Username" required>
        </div>
        <div class="form-group">
            <input type="password" name="password" style="color:#fff
!important;background-color:transparent !important" class="form-control"
placeholder="Your Password" minlength="8" maxlength="30" pattern="(?!.*\d)(?!.*[a-
z])(?!.*[A-Z]).{8,}" title="Must contain at least one number and one uppercase and
lowercase letter, and at least 8 or more characters" required>
        </div>
        <div class="form-group">
            <input type="submit" name="btnsubmit" value="LOGIN" class="btn btn-primary
py-3 px-5">
        </div>
        </form>
        <form action="#" method="post" class="contact-form">
        <div class="form-group">
            <a onclick="getReg()" style="cursor: pointer;color: #e9c05a;">No Account?
Register Now!</a>
        <div id="list"></div>
        </div>
        <div class="form-group">
            <button name="btnforgot" style="background-color: #eac15a08 !important;"
class="btn btn-primary py-3 px-5">Forgot Password</button>
        </div>
        </form>
        </div>
    </div>
</section>
<section class="ftco-section services-section bg-light">
    <div class="container mt-0">
        <div class="row justify-content-center mb-5 pb-3">
            <div class="col-md-7 text-center heading-section ftco-animate">
                <h2 class="mb-4">We Provide Legal Help</h2>

            </div>
        </div>
    </div>

```

```

<div class="row no-gutters d-flex">
  <div class="col-md-3 text-center services align-self-stretch ftco-animate p-4">
    <div class="icon"><span class="flaticon-auction"></span></div>
    <div class="media-body">
      <h3 class="heading mb-3"><a href="#">Get Your Legal Advice</a></h3>
      <p>A small river named Duden flows by their place and supplies it with the
necessary regelialia.</p>
    </div>
  </div>
  <div class="col-md-3 text-center services align-self-stretch ftco-animate p-4">
    <div class="icon"><span class="flaticon-lawyer"></span></div>
    <div class="media-body">
      <h3 class="heading mb-3"><a href="#">Work with Expert Lawyers</a></h3>
      <p>A small river named Duden flows by their place and supplies it with the
necessary regelialia.</p>
    </div>
  </div>
  <div class="col-md-3 text-center services align-self-stretch ftco-animate p-4">
    <div class="icon"><span class="flaticon-money"></span></div>
    <div class="media-body">
      <h3 class="heading mb-3"><a href="#">Have Great Discounted Rates</a></h3>
      <p>A small river named Duden flows by their place and supplies it with the
necessary regelialia.</p>
    </div>
  </div>
  <div class="col-md-3 text-center services align-self-stretch ftco-animate p-4">
    <div class="icon"><span class="ion-ios-help-circle-outline"></span></div>
    <div class="media-body">
      <h3 class="heading mb-3"><a href="#">Review Your Case Documents</a></h3>
      <p>A small river named Duden flows by their place and supplies it with the
necessary regelialia.</p>
    </div>
  </div>
</div>
</div>
</div>
</div>
</section>

```

```

<?php
  require "footer.php";
?>

```

Registration

```
<?php
require "header.php";

if(isset($_POST['btnsubmit']))
{
    $selA="select * from tbl_admin where
admin_username='".$_POST['txtusername']."'";
    $rowA=mysqli_query($con,$selA);
    $numA=mysqli_num_rows($rowA);

    $selE="select * from tbl_station_reg where
station_username='".$_POST['txtusername']."'";
    $rowE=mysqli_query($con,$selE);
    $numE=mysqli_num_rows($rowE);

    $selC="select * from tbl_customer_reg where
customer_username='".$_POST['txtusername']."'";
    $rowC=mysqli_query($con,$selC);
    $numC=mysqli_num_rows($rowC);

    if($numA>0)
    {
        echo "<script>alert('This Username is already exists!')</script>";
    }
    else if($numE>0)
    {
        echo "<script>alert('This Username is already exists!')</script>";
    }
    else if($numC>0)
    {
        echo "<script>alert('This Username is already exists!')</script>";
    }
    else
    {
        $name=strtoupper($_POST['txtname']);
        $address=ucwords($_POST['txtaddress']);
        $gender=$_POST['rdgender'];
        $email=$_POST['txtemail'];
        $phno=$_POST['txtpphno'];
```

```

$dist_id=$_POST['seldistrict'];
$password=$_POST['txtpassword'];
$cpassword=$_POST['txtcpassword'];
$username=strtolower($_POST['txtusername']);
$date=date('Y-m-d');

if($cpassword==$password)
{
    $insertQ="insert into
tbl_customer_reg(customer_name,customer_gender,customer_address,customer_email,customer_phno,
customer_username,customer_password,customer_reg_date,dist_id)
values('".$name."','".$gender."','".$address."','".$email."','".$phno."','".$username."','".$password."','".$date."','".$dist_id."')";

    mysqli_query($con,$insertQ);

    $selectnoti="select * from tbl_customer_reg where
customer_username='".$username."'";
    $rownoti=mysqli_query($con,$selectnoti);
    $datanoti=mysqli_fetch_assoc($rownoti);

    $ntitle="Welcome";
    $ndes="$name Welcome to Crime Investigation Department.";
    $insertNoti="insert into tbl_notification(customer_id,n_title,n_des,status)
values('".$datanoti['customer_id']."','".$ntitle."','".$ndes.','0')";
    mysqli_query($con,$insertNoti);

    header("Location:login.php");
}

else
{
    echo "<script>alert('Confirm your Password!')</script>";
}
}
?>
<section class="hero-wrap hero-wrap-2" style="background-image: url('images/bg_2.jpg');" data-
stellar-background-ratio="0.5">
<div class="overlay"></div>
<div class="container">
<div class="row no-gutters slider-text align-items-end justify-content-center">

```



```

        <select name="seldistrict" id="seldistrict" style="color:#fff
!important;background-color:transparent !important" class="form-control" title="Select
District" required>
            <option value="">----select---</option>
            <?php
            $selQ="select * from tbl_district";
            $row=mysqli_query($con,$selQ);
            while($data=mysqli_fetch_assoc($row))
            {
                ?>
                <option value="<?php echo $data['dist_id']; ?>"
style="color:#000 !important;"><?php echo $data['dist_name']; ?></option>
                <?php
            }
            ?>
        </select>
    </div>
    <div class="form-group">
        <label>Enter your Email</label>
        <input type="email" name="txtemail" style="color:#fff !important;background-
color:transparent !important" class="form-control" placeholder="Your Email"
minlength="4" maxlength="100" class="form-control" pattern="[a-z0-9._%+-]+@[a-z0-
9.-]+\.[a-z]{2,}" title="Enter Email" required>
    </div>
    <div class="form-group">
        <label>Enter your Contact Number</label>
        <input type="tel" name="txtphno" style="color:#fff !important;background-
color:transparent !important" class="form-control" placeholder="Your Contact Number"
minlength="10" maxlength="10" pattern="[0-9]{10}" title="Enter PhoneNumber"
required>
    </div>
    <div class="form-group">
        <label>Create your Username</label>
        <input type="text" name="txtusername" style="text-transform:
lowercase;color:#fff !important;background-color:transparent !important;" class="form-
control" placeholder="Your Username" minlength="4" maxlength="30" pattern="[A-Za-
z0-9]{4,30}" title="Enter Username" required>
    </div>
    <div class="form-group">
        <label>Create your Password</label>
        <input type="password" name="txtpassword" style="color:#fff
!important;background-color:transparent !important" class="form-control"
placeholder="Your Password" minlength="8" maxlength="30" pattern="(?=.*\d)(?=.*[a-
z])(?=.*[A-Z]).{8,}" title="Must contain at least one number and one uppercase and
lowercase letter, and at least 8 or more characters" required>
    </div>

```



```

        <div class="form-group">
            <label>Confirm your Password</label>
            <input type="password" name="txtcpassword" style="color:#fff
!important;background-color:transparent !important" class="form-control"
placeholder="Confirm Your Password" minlength="8" maxlength="30"
pattern="(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{8,}" title="Must contain at least one number and one
uppercase and lowercase letter, and at least 8 or more characters" required>
        </div>
        <div class="form-group">
            <input type="submit" name="btnsubmit" value="Register" style="background-color:
#eac15a08 !important;" class="btn btn-primary py-3 px-5" minlength="8" maxlength="30"
pattern="(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{8,}" title="Must contain at least one number and one
uppercase and lowercase letter, and at least 8 or more characters" required>
        </div>
    </form>
</div>
    <div class="col-md-2"></div>
</div>
</div>
</section>

<?php
    require "footer.php";
?>

```

5.3 DEBUGGING

Debugging is the process of detecting and removing of existing and potential errors (also called as ‘bugs’) in a software code that can cause it to behave unexpectedly or crash. To prevent incorrect operation of a software or system, debugging is used to find and resolve bugs or defects. When various subsystems or modules are tightly coupled, debugging becomes harder as any change in one module may cause more bugs to appear in another. Sometimes it takes more time to debug a program than to code it. To debug a program, user has to start with a problem, isolate the source code of the problem, and then fix it. A user of a program must know how to fix the problem as knowledge about problem analysis is expected. When the bug is fixed, then the software is ready to use. Debugging tools (called debuggers) are used to identify coding errors at various development stages. They are used to reproduce the conditions in which error has occurred, then examine the program state at that time and locate the cause. Programmers can trace the program execution step-by-step by evaluating the value of variables and stop the execution wherever required to get the value of variables or reset the program variables. Some programming language packages provide a debugger for checking the code for errors while it is being written at run time.

5.4 UNIT TESTING

The unit test phase entails converting the design language in to program code and, most important, designing and carrying out tests of the individual units. Once individual modules or units have been tested and accepted, the integration and test phase begin. This initial part of structural testing corresponds to some quick checks that a developer performs before subjecting the code to more extensive code coverage testing or code complexity testing. The developer can perform certain obvious tests knowing the input variables and the corresponding expected output variables. This can be a quick test that checks out any obvious mistakes. This can even be done prior to formal reviews of static testing so that the review mechanism does not waste time.

Unit testing is undertaken when a module has been created and successfully reviewed. In order to test a single module, we need to provide a complete environment i.e. besides the module we would require,

- The procedures belonging to other modules that the module under test calls o and non-local data structures that module accesses.
- A procedure to call the functions of the module under test with appropriate parameters.

In my project each module is separated and tested. That means in the admin side, patient side and therapist side are separately tested. Check the duplication of data and the duplication is removed. And ensure that the updating is recorded correctly.

SYSTEM PLANNING AND SCHEDULING

6.1 INTRODUCTION TO SYSTEM PLANNING AND SCHEDULING

Planning and scheduling are distinct but inseparable aspects of managing the successful project. The process of planning primarily deals with selecting the appropriate policies and procedures in order to achieve the objectives of the project. Scheduling converts the project action plans for scope, time cost and quality into an operating timetable. The translating of the project criteria for scope, time, cost, and quality and the requirements for human resources, communications, risk and procurement into workable “machinery” for the project team a critical interface juncture for the project team. Taken together with the project plan and budget, the schedule becomes the major tool for the management of projects. In addition, the integrated cost-time schedule serves as the fundamental basis for monitoring and controlling project activity throughout its life cycle.

This basic level paper addresses the integrated processes of planning and scheduling of multifaced/multidisciplinary programs. The paper presents a working level summary of the major Project Management topics involved in the planning process. The paper also details a systematic process for transforming the Project Plan into the Schedule and the use of the Project Schedule as a model for project control. Intended for the project management novice, the paper concludes with a suggested professional development scheme.

6.1.1 PLANNING A SOFTWARE PROJECT

The basic project planning steps that every project manager needs to know can be broken down as parts of the first two phases of project management: Initiation and Planning. While those phases give a broad outline of what should be happening at different stages of a project’s lifecycle, they don’t provide much of a clear picture of how to go about your project planning.

6.1.1.1 STEPS INVOLVED IN PLANNING A SYSTEM

1. Create and Analyse Business Case
2. Identify and Meet Stakeholders for Approval
3. Define Project Scope
4. Set Project Goals and Objectives
5. Determine Project Deliverables
6. Create Project Schedule and Milestones
7. Assignment of Tasks
8. Carry Out Risk Assessment

Project planning doesn't have to be difficult or cause any nervous stress since the beginning of every project is basically the same. You can follow the same set project planning steps and hone them through experience of every project you are involved with.

Breaking down the steps

1. How to Create and Analyse Business Case?

The business case is the reason why your organization needs to carry out the project. It should outline the problem, such as a lack of repeat customers or a day longer supply line than competitors and describe how this will be solved and how much monetary benefit should accrue to the organization once the project is completed.

2. How to Identify and Meet Relevant Stakeholders for Approval?

Identifying project stakeholders means listing anyone who will be affected by your project, so includes the public and government regulatory agencies. For the project planning phase however, it should only be necessary to meet those who will directly decide whether the project will happen or not.

3. Define Project Scope?

The scope of your project is an outline of what it is and isn't setting out to achieve. It is necessary to delineate the boundaries of your project to prevent "scope creep", i.e., your resources going towards something that's not in your project's goals.

4. Set Goals and Objectives

The goals and objectives for your project will build on the initial objectives outlined in the business plan. At this step you will give finer detail to the initial broad ideas and set them in a project charter as reference points for your project as it proceeds.

5. Determine Deliverables

Deliverables are the concrete results that your project produces. One of the most important project planning steps is to decide on what these deliverables will be and who is responsible for both producing and receiving them.

6. Create Project Schedule and Milestones

Your project schedule is a very important document that outlines when different tasks of a project are due to begin and end, along with major measurement milestones. It will be referred to when measuring project progress. It will be available to all stakeholders and should be adhered to as closely as possible.

7. Assignment of Tasks

Within your team everyone should know what their role is and who is responsible for different elements of the project. Assigning tasks clearly should remove any uncertainty about roles and responsibilities on your team.

8. Carry Out Risk Assessment

Having a functional risk management plan means performing a strong assessment at the planning stage of the project. All potential risks should be identified along with their possible effect on the project and likelihood of occurring.

SYSTEM COST ESTIMATION

7.1. INTRODUCTION

A project can only come together with all the necessary materials and labour, and those materials and labours cost money. Putting together a budget that keeps costs to a minimum, while maximizing the project's quality and scope can be challenging. This is why proper cost estimation is important.

Cost estimation in project management is the process of forecasting the financial and other resources needed to complete a project within a defined scope. Cost estimation accounts for each element required for the project—from materials to labour—and calculates a total amount that determines a project's budget. An initial cost estimate can determine whether an organization greenlights a project, and if the project moves forward, the estimate can be a factor in defining the project's scope. If the cost estimation comes in too high, an organization may decide to pare down the project to fit what they can afford (it is also required to begin securing funding for the project). Once the project is in motion, the cost estimate is used to manage all of its affiliated costs in order to keep the project on budget.

There are two key types of costs addressed by the cost estimation process:

1. **Direct costs:** Costs associated with a single area, such as a department or the project itself. Examples of direct costs include fixed labour, materials, and equipment.
2. **Indirect costs:** Costs incurred by the organization at large, such as utilities and quality control.

7.2 FUNCTIONAL POINT BASED ESTIMATION

The basic and primary purpose of the functional point analysis is to measure and provide the software application functional size to the client, customer, and the stakeholder on their request. Further, it is used to measure the software project development along with its maintenance, consistently throughout the project irrespective of the tools and the technologies.

Allan J. Albrecht initially developed function Point Analysis in 1979 at IBM and it has been further modified by the International Function Point Users Group (IFPUG). FPA is used to make estimate of the software project, including its testing in terms of functionality or function size of the software product. However, functional point analysis may be used for the test estimation of the product. The functional size of the product is measured in terms of the function point, which is a standard of measurement to measure the software application. FPs of an application is found out by counting the number and types of functions used in the applications.

SYSTEM MAINTENANCE

9.1. INTRODUCTION FOR SYSTEM 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.

9.2 MAINTENANCE

In my project Catering Management System there are three types of maintenance:

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

Corrective maintenance:

removes software faults.

Perfective maintenance:

improves the system without changing its functionality. The objective of perfective maintenance should be to prevent failures and optimize the software.

Adaptive maintenance:

modifies the software to keep it up to date with its operative environment. It may be needed because of changes in the user requirements, changes in target platform, or changes in external interfaces. Minor adaptive changes should be handled by normal maintenance process. Major adaptive changes should be carried out as a separate development project.

FUTURE ENHANCEMENT AND SCOPE OF FURTHER DEVELOPMENT

11.1 MERITS OF THE SYSTEM

Currently there is no online application available to report crime online. In order to report any complaints related with crime, people have to contact nearest police station. People of the particular city aren't aware of crime related things such as list of most wanted criminals of their city, latest crime related news, missing person of their area etc. people have to view news channels or read newspapers for such crime related information's. Thus, we can say that existing system is manual and does not provide all the information from one source. malpractices. The new system is very faster and efficient than the older one.

- The user interface of this app is user-friendly
- functionality is understandable to the users
- This app gives a complete picture of emotional state of patient
- Provide task to improve their mental health condition
- Cost effective
- Time saving

11.2 LIMITATIONS OF THE SYSTEM

This system using electronic documents which is easy to modify or hack, if any attacker breaks into the system, he can access all the files and data that are related to the online crime reporting management system.

11.3 FUTURE ENHANCEMENT OF THE SYSTEM

Anything cannot be ended in a single step. It is the fact that nothing permanent in this world. So, this utility also has some future enhancements in the evergreen and booming IT industry. Change is inevitable. The project entitled " Crime Management System " was successfully designed developed and tested. The system and the architecture are a compatible one, so addition of new modules can be done without much difficulty. Since this module has its unique properties, it can extend further to make this system a complete one. Scope of further development the various future enhancement which shall include in this software are:

- The method of video conferencing can be added to make the project more lively.
- Develop Admin profile, allowing replaying to each user's query or suggestions.
- Make the application more secure.
- An intercom facility will add a little more flexible communication between the Admin and Police Station in charge, Admin and Customer etc.