ONLINE CRIME REPORT SYSTEM

A PROJECT REPORT

Submitted by

DEEPANSHU
2100290140056
SAGAR KUMAR
2100290140115
SHIVANI PANDEY
2100290140130

Submitted in partial fulfilment of the
Requirements for the Degree of

MASTER OF COMPUTER APPLICATION

Under the Supervision of
Dr. Sangeeta Arora
Associate Professor

Submitted to

DEPARTMENT OF COMPUTER APPLICATIONS
KIET Group of Institutions, Ghaziabad
Uttar Pradesh-201206
(JUNE 2023)

CERTIFICATE

Certified that **Deepanshu 2100290140056, Sagar Kumar 2100290140112, Shivani Pandey 2100290140130** has/have carried out the project work having "**Online Crime Report System"** for Master of Computer Applications from Dr. A.P.J. Abdul Kalam Technical University (AKTU) (formerly UPTU), Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself / herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Date:

Deepanshu - 2100290140056

Sagar Kumar - 2100290140112

Shivani Pandey - 2100290140130

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date:

Dr. Sangeeta Arora
Associate Professor
Department of Computer Applications
KIET Group of Institutions, Ghaziabad

Signature of Internal Examiner **Signature of External Examiner**

Dr. Arun Tripathi
Head, Department of Computer Applications
KIET Group of Institutions, Ghaziabad

ABSTRACT

We have designed an online crime report project in java, which has multiple roles i.e., Admin and User. Criminal records include the identity of people like photos etc. and cases registered against them. Admin being the primary user. Admin can add/remove/update any details related to the system and keep the records of the criminals. Admin can manage registered FIRs and other documentation.

Users can register and log in to the system. Users will be able to contact the police stations, report crimes, file FIRs, etc.

An online crime report system is a web-based platform that allows citizens to report crimes to the police electronically. This system provides an efficient and convenient way for individuals to report crimes from the comfort of their homes or offices, without the need to physically visit a police station.

The system typically requires users to provide details about the crime, such as the location, date and time, and a description of the incident. Users can also upload supporting documents, such as photos, to aid in the investigation process.

ACKNOWLEDGEMENTS

Success in life is never attained single-handedly. My deepest gratitude goes to my thesis supervisor, **Dr. Sangeeta Arora** for her guidance, help and encouragement throughout my research work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to **Dr. Arun Tripathi**, Professor and Head, Department of Computer Applications, for his insightful comments and administrative help at various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to **my all-faculty** members & also my family members and all those who have directly and indirectly provided me with moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

Deepanshu

Sagar Kumar

Shivani Pandey

TABLE OF CONTENTS

List of Chapters

Chapter **Page No**

LIST OF FIGURES

Chapter 1

Introduction

An online crime report system is a web-based platform that allows individuals to report crimes or suspicious activities to law enforcement agencies via the internet. This system provides a simple and convenient way for people to report crimes without the need to physically visit a police station.

The online crime report system typically requires users to provide information about the crime they are reporting, such as the location, time and date, and a description of the incident. Users may also be asked to provide personal information, such as their name and contact details, to enable law enforcement agencies to follow up on the report.

The benefits of an online crime report system include faster response times, increased efficiency, and improved communication between law enforcement agencies and the public. Additionally, it allows law enforcement agencies to better allocate their resources and prioritize cases based on the severity of the crime.

However, it is important to note that an online crime report system may not be suitable for all types of crimes, especially in cases where immediate intervention is required. Therefore, it is important to understand the limitations of such a system and to use it appropriately.

1.1 Project Description

We have designed an online crime report project in java, which has multiple roles i.e., Admin and User. Criminal records include the identity of people like photos etc. and cases registered against them. Admin being the primary user. Admin can add/remove/update any details related to the system and keep the records of the criminals. Admin can manage registered FIRs and other documentation.

Users can register and log in to the system. Users will be able to contact the police stations, report crimes, file FIRs, etc.

An online crime report system is a web-based platform that allows citizens to report crimes to the police electronically. This system provides an efficient and convenient way for individuals to report crimes from the comfort of their homes or offices, without the need to physically visit a police station.

The system typically requires users to provide details about the crime, such as the location, date and time, and a description of the incident. Users can also upload supporting documents, such as photos, to aid in the investigation process.

1.2 Project Scope

The output of an online crime report system is typically in the form of a report that contains information about the crime that has been reported. The report may include details such as the location, time and date of the incident, a description of the crime, and the personal details of the person who reported it.

In addition to the report itself, the online crime report system may generate other outputs such as notifications to law enforcement agencies about the reported crime, alerts to relevant authorities or emergency services, and updates to the person who reported the crime about the status of their report.

The system may also generate statistics and other data that can be used to analyze crime trends and patterns, which can help law enforcement agencies to identify areas of high crime activity and allocate resources accordingly.

Overall, the output of an online crime report system is designed to provide a more efficient and effective way for people to report crimes and for law enforcement agencies to respond to them, ultimately leading to a safer and more secure community.

1.3 Hardware & Software Requirement:

Hardware Interfaces Requirement

- Processor: i3 11^{th} Gen
- RAM: 8 GB
- Operating System: Windows 11
- System Type: 64-bit operating system
- Hard disk: 256 GB

Software Interfaces Requirement

- Eclipse
- MySQL

- VS CODE
- Apache Tomcat Server

All these types of software automatic configure inside operating system after installation it is having Java, MYSQL, Apache and operating system base configuration file, it doesn't need to configure manually.

Chapter 2

Feasibility Study

2.1 Operational Study

This project will be developed on computer, so first check whether the technology is technically available or not. Now a day's computer interaction with any job becomes common for any kind of job or work.

And because of increasing usage of Computer, Computer is also available with a variety of hardware. Vendors can fulfill any type of hardware requirement. The whole project is developed by some special tools or by using languages and databases, which are also available in a variety.

Preliminary investigation of a system examines the feasibility of a system that is useful to an organization. It is the first phase of system development.

The main objective of this phase is to identify the current deficiencies in the user's environment and to determine which existing problems are going to be solve in proposed system and also which new function needs to be added in proposed system.

An important outcome of such preliminary investigation is to determine whether the system that will meet all needed requirements.

Thus, three tests are carried out on the system namely operation, technical and economical.

Any project is beneficial if and only satisfies the organization requirement. For any new system setup, it only meets to be communicated and work the other supporting system.

The new system meets all existing operations since it provides right information at a right time to the right user. A Leigh man can easily operate with the system.

2.2 Technical Feasibility

Technical Feasibility examines whether the technology needed is available and if it is available then it feasible to carry out all project activities.

The technical needs of a system include:

- The facility to produce outputs in each time.
- Ability to process large number of transactions at a particular speed.
- Giving response to users under certain conditions.

The technology needed for our system is mainly:

- Latest version of browsers.
- Any operating system.

These technologies are available which helps to carry out the system efficiently.

2.3 Economic Feasibility

Economic feasibility of a system examines whether the finance is available for implementing the new system and whether the money spent is recoverable the satisfaction.

The cost involves is in designing and developing a good investment for the organization.

Thus, hardware requirements used for proposed system are very standard. Moreover, by making use of the proposed system to carry out the work speedily will increase and save the valuable time of an organization.

In the proposed system the finance is highly required for the installation of the software's which can also be recovered by implementing a better system.

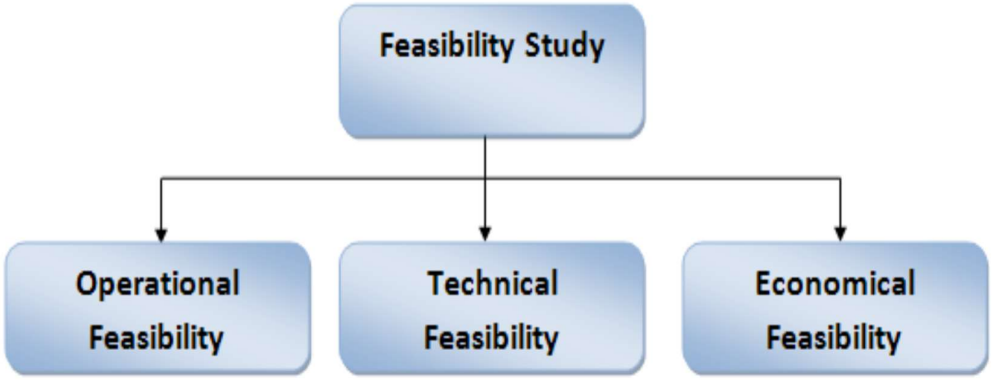

Fig. 2.1 Feasibility Study

Chapter 3

Database Design

3.1 Database Table

3.1.1 User Table

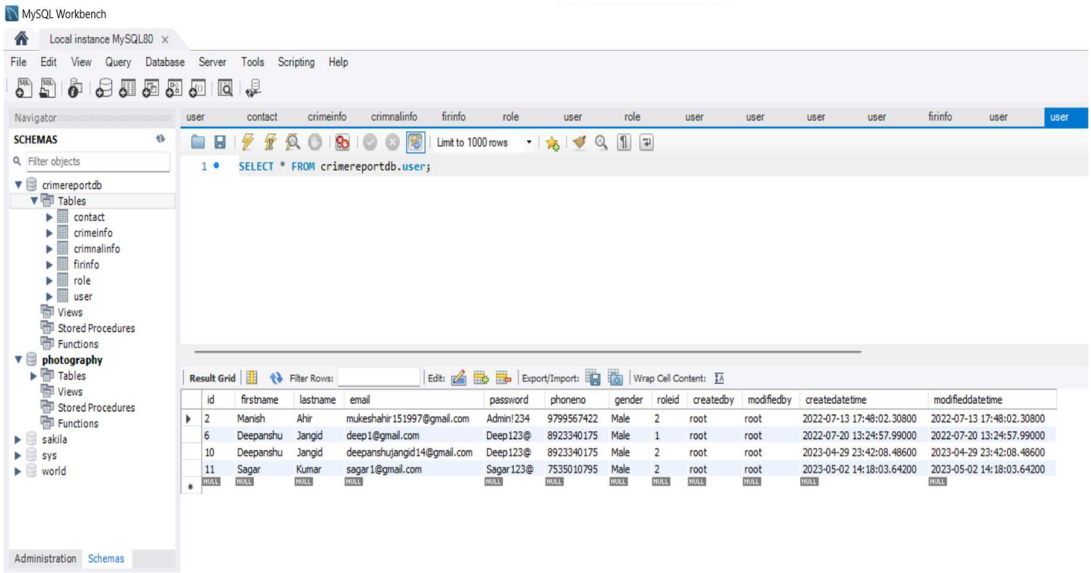

Fig. 3.1 User Table

In a major project involving an online crime report system, a user database table would typically be designed to store information about the system's users. The user database table would include various fields or columns to capture relevant details about each user. Here's a description of the common fields you might consider including in your user database table:

1. User ID: A unique identifier assigned to each user in the system. This field serves as the primary key for the table.
2. Username: The chosen username or login name of the user. This field helps identify and authenticate users.
3. Password: The encrypted or hashed password associated with the user's account. It should be stored securely to protect user data.

4. Email Address: The user's email address, which can be used for communication purposes or account verification.
5. First Name: The user's first name or display name. This field allows for personalized interactions within the system.
6. Last Name: The user's last name or display name. This field allows for personalized interactions within the system.
7. Gender: The user's gender. This field allows for personalized interactions within the system.
8. Contact Number: The user's contact number, which can be useful for communication or emergency contact purposes.
9. Address: The user's residential or mailing address. This field can be helpful for location-based services or for law enforcement purposes.

3.1.2 Crime Info Table

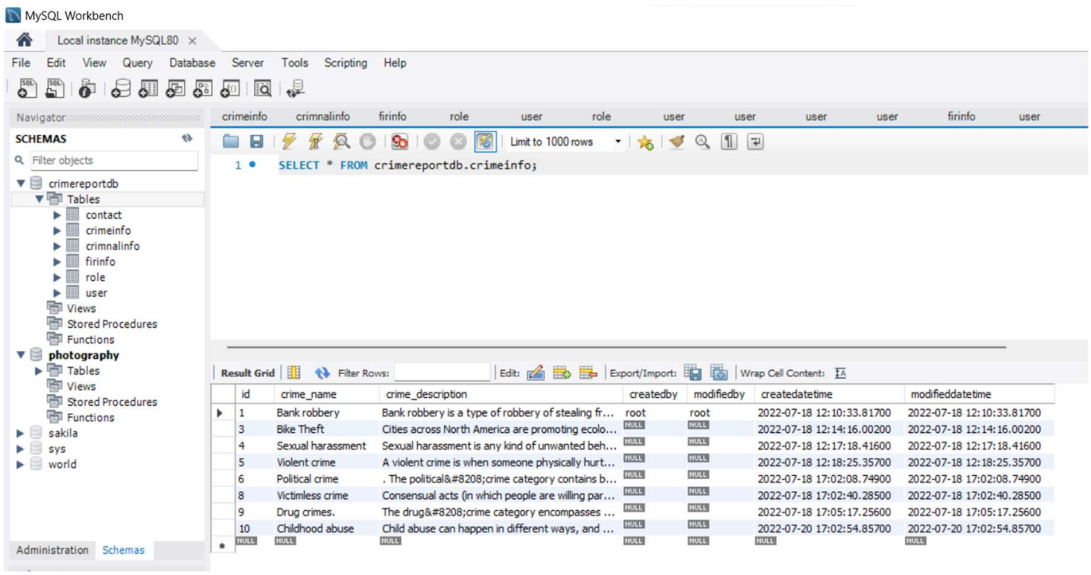

Fig. 3.2 Crime Info Table

In a major project involving an online crime report system, a crime information database table would be designed to store relevant details about reported crimes. The crime info database table would include various fields or columns to capture essential information related to each crime. Here's a description of the common fields you might consider including in your crime info database table:

1. Crime ID: A unique identifier assigned to each crime in the system. This field serves as the primary key for the table.
2. Date and Time of Occurrence: The date and time when the crime occurred or was reported.
3. Location: The location where the crime took place, including address details or geographic coordinates.

4. Description: A detailed description of the crime, including the nature of the offense, modus operandi, and any other relevant information.
5. Offenses: Details of the offenses committed in the crime, including the specific criminal acts or violations involved.
6. Suspects/Persons of Interest: Information about individuals who are suspected or considered persons of interest in connection with the crime. This field can include their names, descriptions, and any available identifying information.
7. Witnesses: Details about individuals who witnessed the crime, including their names, contact details, and statements.
8. Evidence/Attachments: A field to store any evidence or supporting documents related to the crime, such as photographs, videos, or documents.

3.1.3 Criminal Info Table

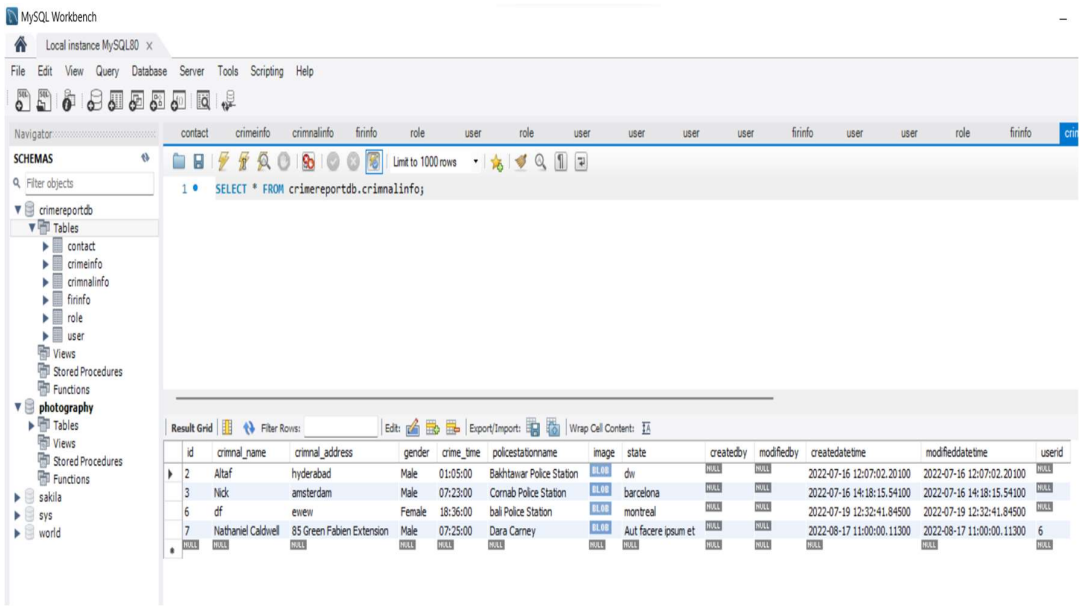

Fig. 3.3 Criminal Info Table

In a major project involving an online crime report system, a criminal information database table would be designed to store relevant details about criminals and their associated information. The criminal info database table would include various fields or columns to capture essential data about each criminal. Here's a description of the common fields you might consider including in your criminal info database table:

1. Criminal ID: A unique identifier assigned to each criminal in the system. This field serves as the primary key for the table.
2. Full Name: The full name of the criminal, including their first name, middle name (if applicable), and last name.
3. Date of Birth: The date of birth of the criminal, which can help in determining their age and other demographic information.

4. Gender: The gender of the criminal, which can be recorded as male, female, or other.
5. Nationality: The nationality or country of origin of the criminal.
6. Addresses: The addresses associated with the criminal, including current and past addresses, if available.
7. Criminal Offenses: A field to record the criminal offenses committed by the individual. This field can include multiple offenses, each with details such as offense type, date, and severity.
8. Photographs: A field to store photographs or mugshots of the criminal, which can aid in identification.

3.1.4 FIR INFO Table

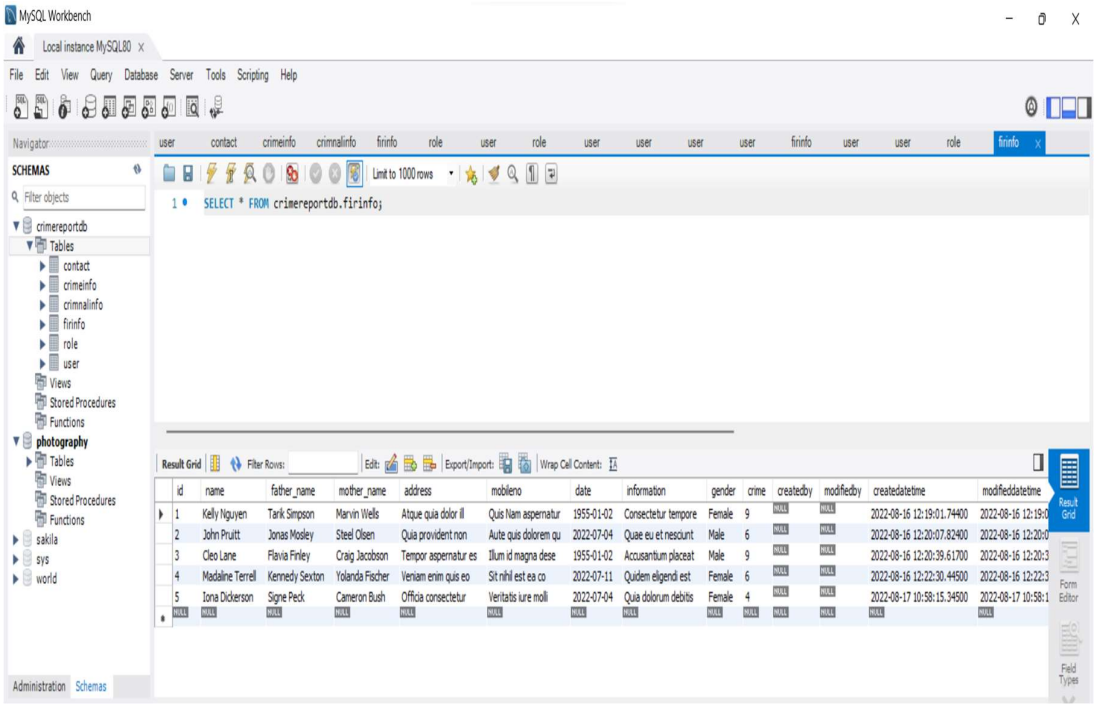

Fig. 3.4 FIR Table

In a major project involving an online crime report system, an FIR (First Information Report) info database table would be designed to store relevant details about registered FIRs. The FIR info database table would include various fields or columns to capture essential information related to each FIR. Here's a description of the common fields you might consider including in your FIR info database table:

1. FIR ID: A unique identifier assigned to each FIR in the system. This field serves as the primary key for the table.
2. Date and Time of Registration: The date and time when the FIR was registered. This field helps in tracking the timeline of the reported incident.

3. Complainant Name: The name of the person who filed the FIR or lodged the complaint.
4. Complainant Contact Details: Contact information of the complainant, including phone number and email address, for communication purposes.
5. Incident Date and Time: The date and time when the incident occurred or was reported.
6. Incident Location: The location where the incident took place, including address details or geographic coordinates.
7. Incident Description: A detailed description of the incident as provided by the complainant or the reporting party.
8. Assigned Station/Department: The police station or law enforcement department to which the FIR is assigned.

3.1.5 Contact Table

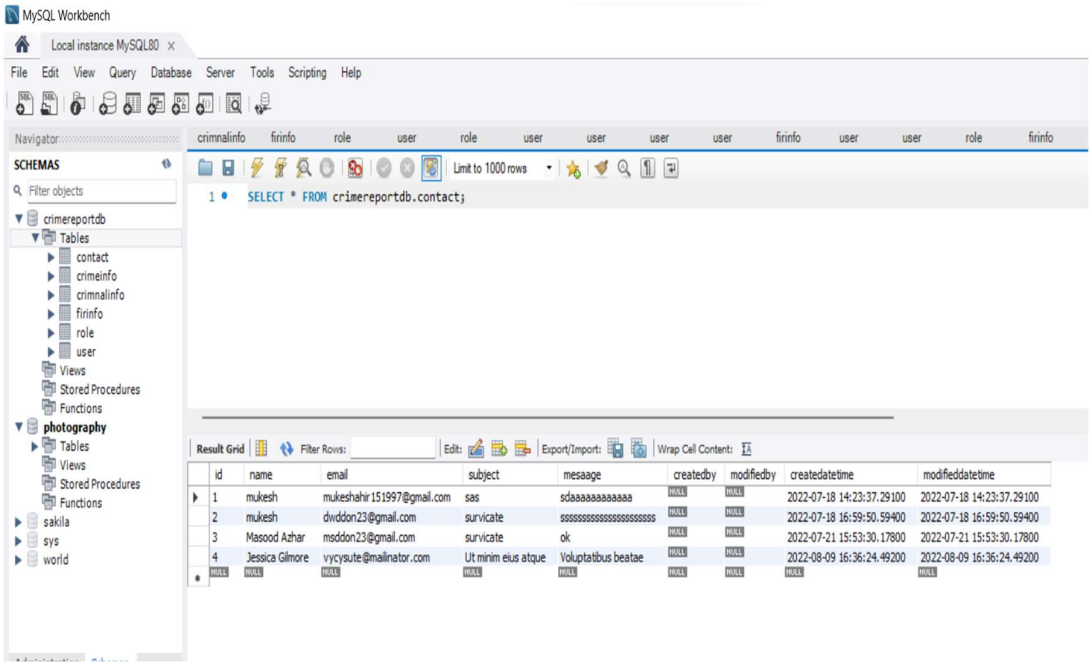

Fig. 3.5 Contact Table

In a major project, a contact database table can be designed to store information about contacts or individuals associated with the system. The contact database table would include various fields or columns to capture relevant details about each contact. Here's a description of the common fields you might consider including in your contact database table:

1. Contact ID: A unique identifier assigned to each contact in the system. This field serves as the primary key for the table.
2. Full Name: The full name of the contact, including their first name, middle name (if applicable), and last name.

3. Email Address: The email address of the contact, which can be used for communication purposes.
4. Phone Number: The phone number of the contact, which can be useful for direct communication.
5. Address: The residential or office address of the contact. This field can help in storing and retrieving location-related information.

3.2 Use Case Diagram

Use Case Diagram for User and Authorization

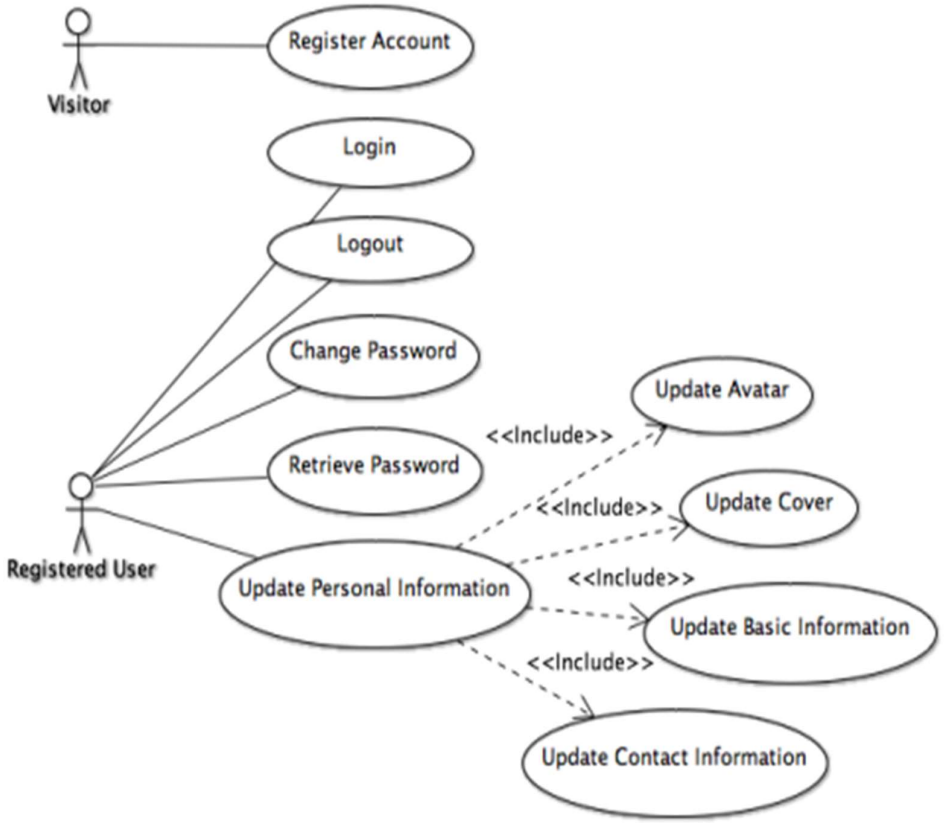

Fig. 3.6 Use Case Diagram

3.3 Sequence Diagram

3.3.1 For Login

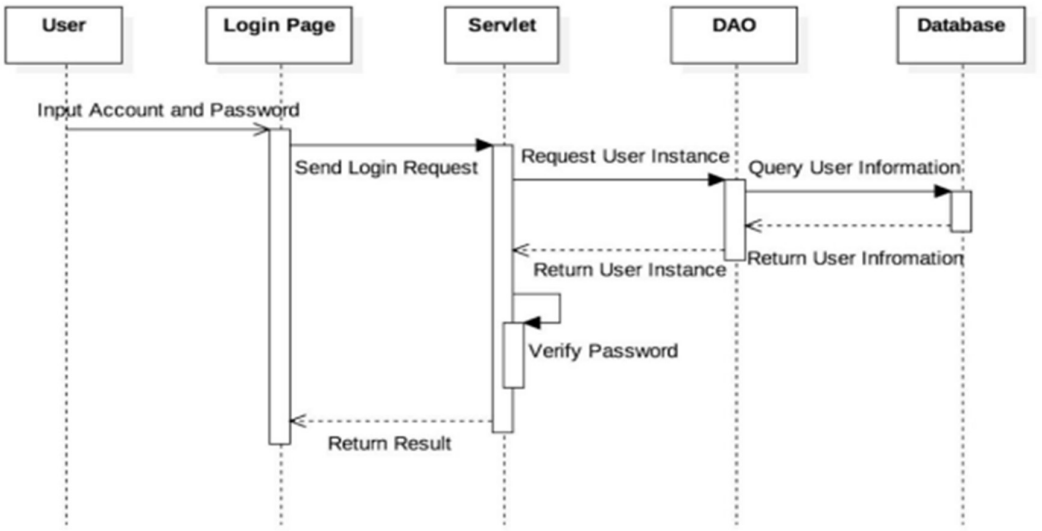

Fig. 3.7 Sequence Diagram-Login

The basic flow of user login is as follows:

1. The user accesses the login page and input his account and password.

2. The login page sends the login request together with the user's input information to the LoginServlet

3. LoginServlet calls the user DAO (data access object) to perform query operation on the database, and returns the instance object of user

4. loginByUserEmail method Compare and verify the input password with the correct password (saved in the instance object).

5. If the password is wrong, login fails and then returns to the login page, reporting an error. Otherwise, login succeeds and then jumps to the user's personal homepage.

3.3.2 Sequence Diagram for Personal Information

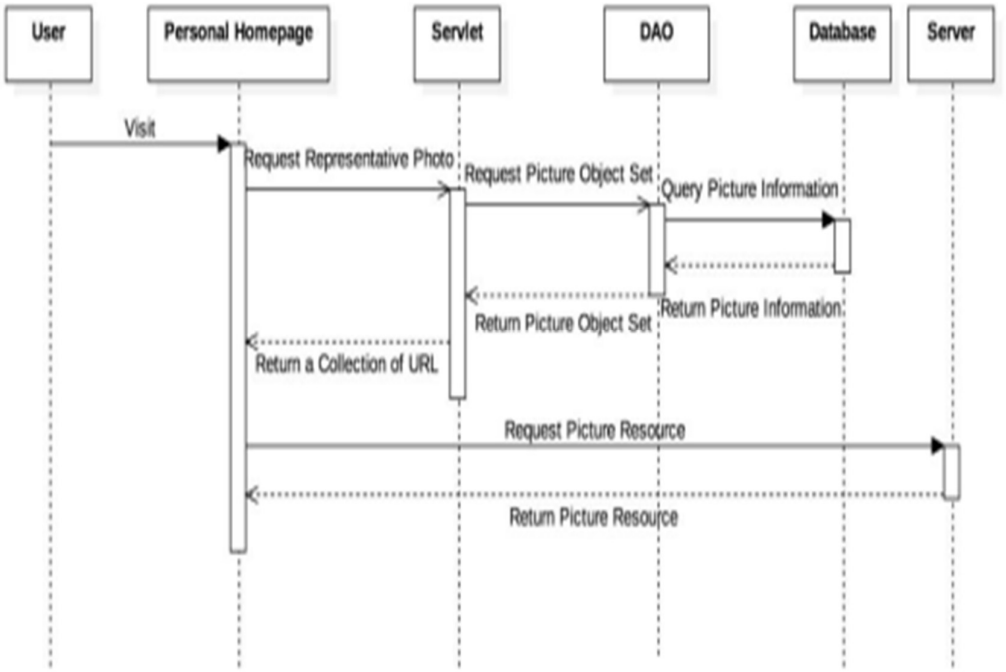

Fig. 3.8 Sequence Diagram-Personal Information

The basic flow of displaying a user's representative photos is as follows:

1.The user visits the personal homepage for browsing.

2.Personal homepage requests the HomepageServlet for relevant information about the current user's representative photos.

3.HomepageServlet calls the user data access object for query operation and returns a collection of picture objects containing information of the user's representative photos.

4.The Personal homepage requests to the server, accesses picture resources and displays representative photos using the URLs.

3.4 Data Flow Diagram (DFD)

Level 1 DFD ADMIN

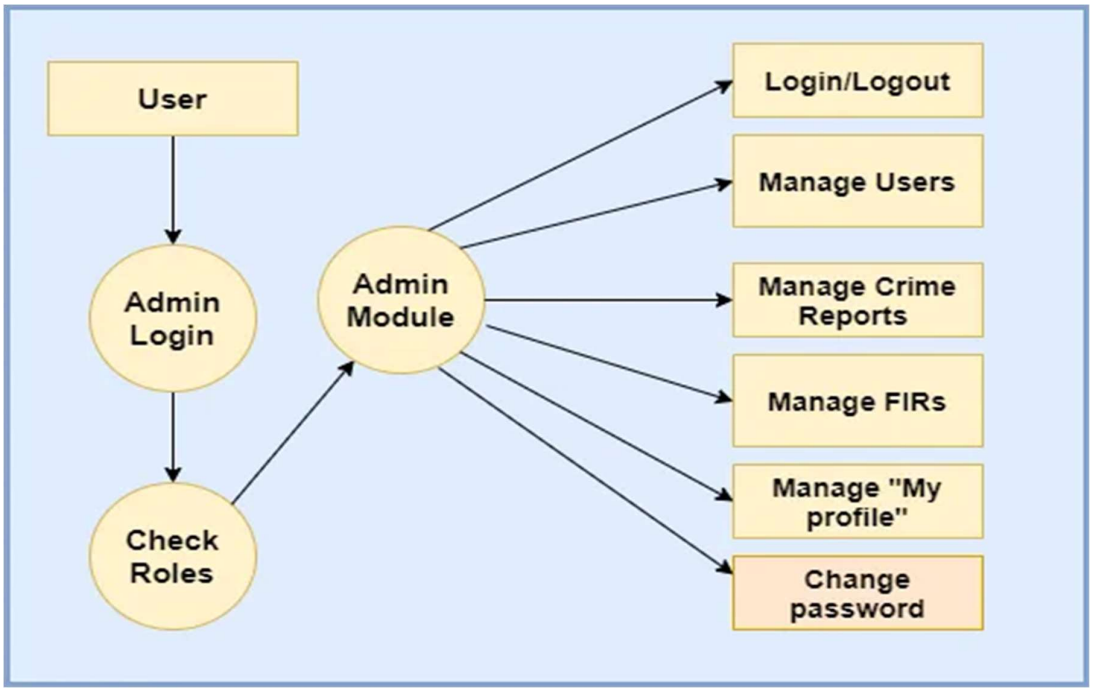

Fig. 3.9 DFD Admin

The DFD-ADMIN (Data Flow Diagram - ADMIN) is a critical component of our project, designed to illustrate the flow of data and processes within the administrative functions of our system. It provides a visual representation of how information is captured, processed, and stored within the administrative module of our project.

The DFD-ADMIN helps to provide a high-level understanding of how data flows and processes are interconnected within the administrative module of our project. It aids in identifying potential bottlenecks, data dependencies, and areas for improvement within the administrative functions.

By visually representing the data flow and processes, the DFD-ADMIN assists in communicating the overall architecture and functionality of the administrative module to stakeholders, project team members, and developers. It serves as a blueprint for designing and implementing the administrative functionalities effectively and ensures a clear understanding of how the system operates.

Level 2-DFD USER

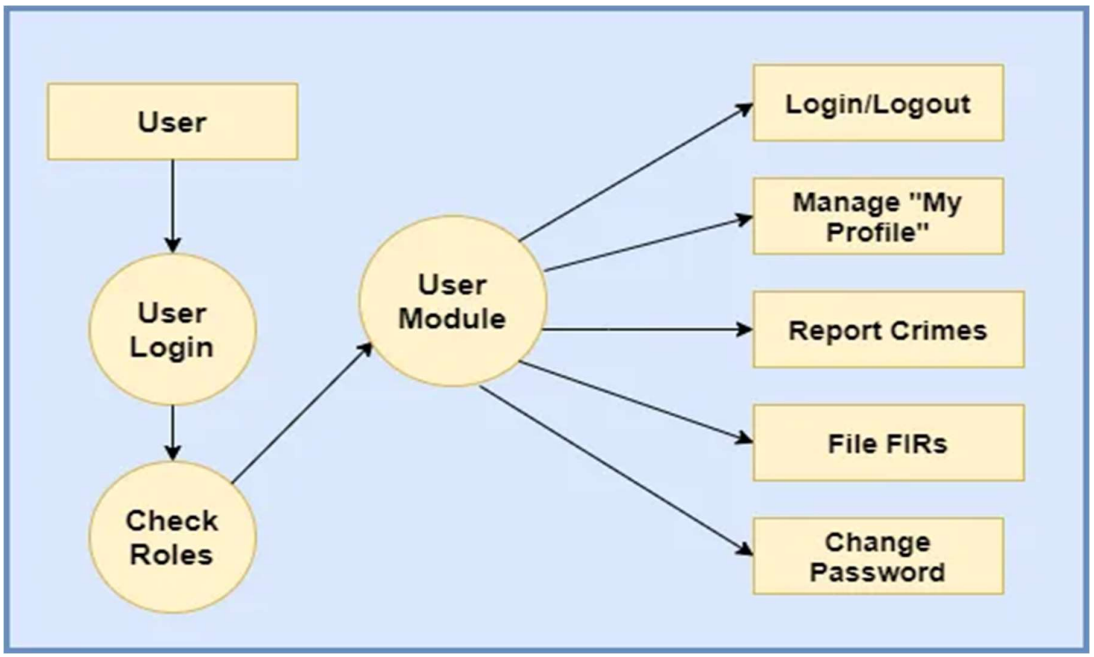

Fig. 3.10 DFD User

The DFD-USER (Data Flow Diagram - USER) is an important component of our project, designed to depict the flow of data and processes related to user interactions within the system. It provides a visual representation of how data is exchanged between users and the various components of our project.

The DFD-USER provides a visual representation of how users interact with the system, illustrating the flow of data and processes involved in their interactions. It helps in understanding the overall architecture and functionality from a user's perspective.

By using the DFD-USER, stakeholders, project team members, and developers can gain a clear understanding of how data flows and processes are interconnected within the user interactions. It assists in identifying user requirements, potential bottlenecks, and areas for improvement in the user experience.

3.5 E-R Diagram

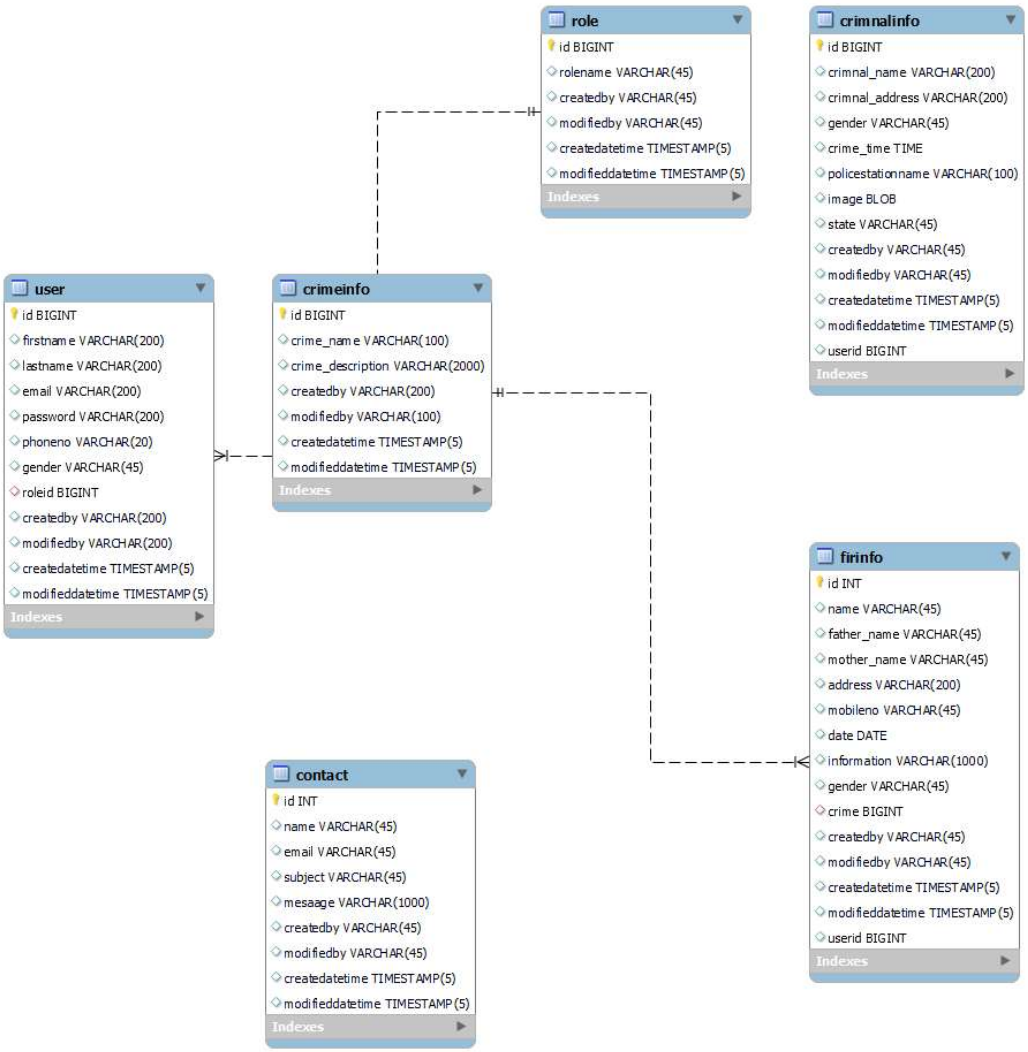

Fig. 3.11 E-R Diagram

Chapter 4

Project Module's

4.1 Admin Module

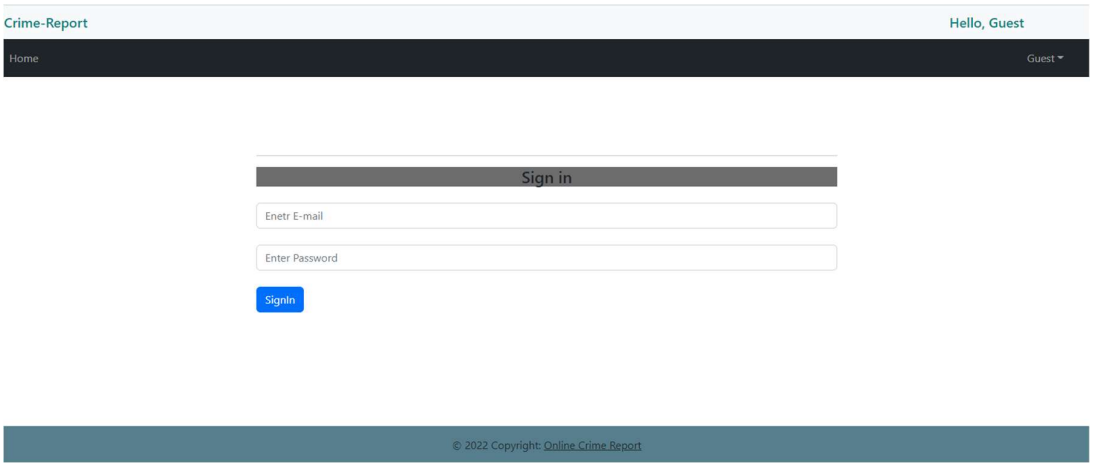

Fig. 4.1 Admin Module

Our Login/Signup Page Module is an essential component of our project, designed to provide a seamless and secure user authentication experience. This module allows users to create new accounts by registering with their personal information and credentials or log in using their existing account details.

The module features a user-friendly interface with clear and intuitive design elements, ensuring that users can easily navigate through the login/signup process. It includes fields for entering email addresses, passwords, and other necessary information, with appropriate validation to ensure data accuracy and security.

4.2 Criminal Module.

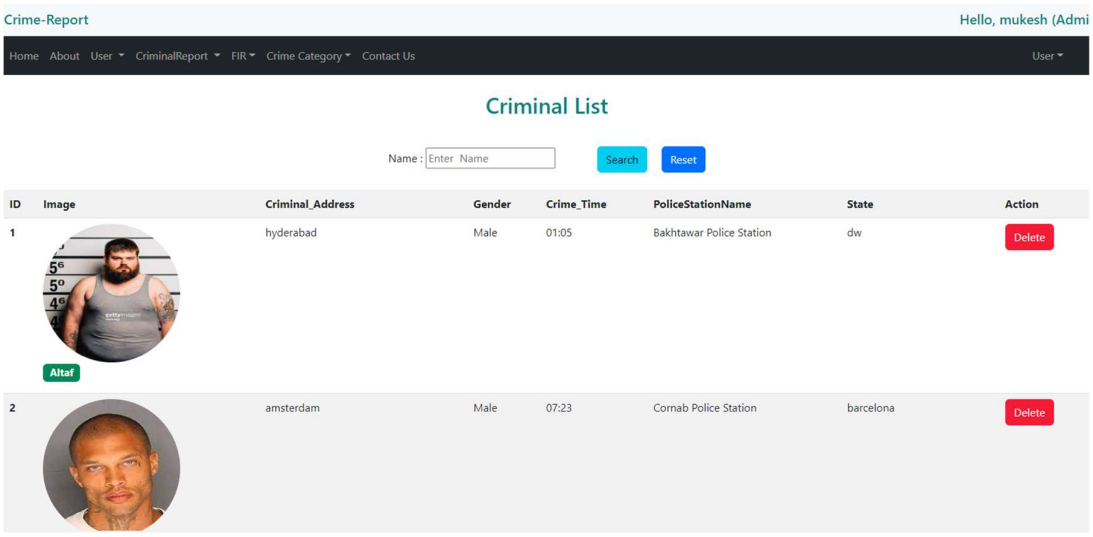

Fig. 4.2 Criminal Module

Our Criminal List Page Module is a crucial component of our project, designed to facilitate the management and display of criminal records in an organized and accessible manner. This module serves as a comprehensive database that allows authorized users to search, view, and update criminal information efficiently.

The module features a user-friendly interface with a clean and intuitive design, making it easy for users to navigate and locate relevant criminal records. It provides a search functionality that enables users to search for criminals based on various criteria such as name, age, gender, and offense type.

4.3 FIR Module

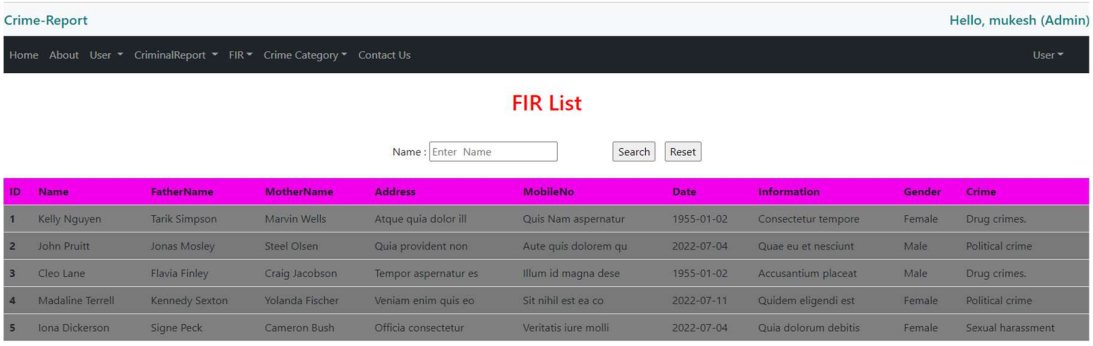

Fig. 4.3 FIR Module

Our FIR (First Information Report) Module is a vital component of our project, designed to facilitate the efficient and accurate recording and management of FIRs. This module serves as a comprehensive system that enables authorized personnel, such as law

enforcement officers or administrators, to document and handle reported crimes effectively.

The module features a user-friendly interface with intuitive design elements, making it easy for users to navigate and enter relevant information. It includes a form that captures essential details related to the incident, such as the complainant's information, date and time of the incident, location, description of the crime, and any supporting evidence or witnesses.

Upon submitting an FIR, the module generates a unique reference number and timestamps the entry for future reference. This reference number allows users to track and retrieve the specific FIR easily, ensuring efficient management and retrieval of records.

4.4 Crime Module

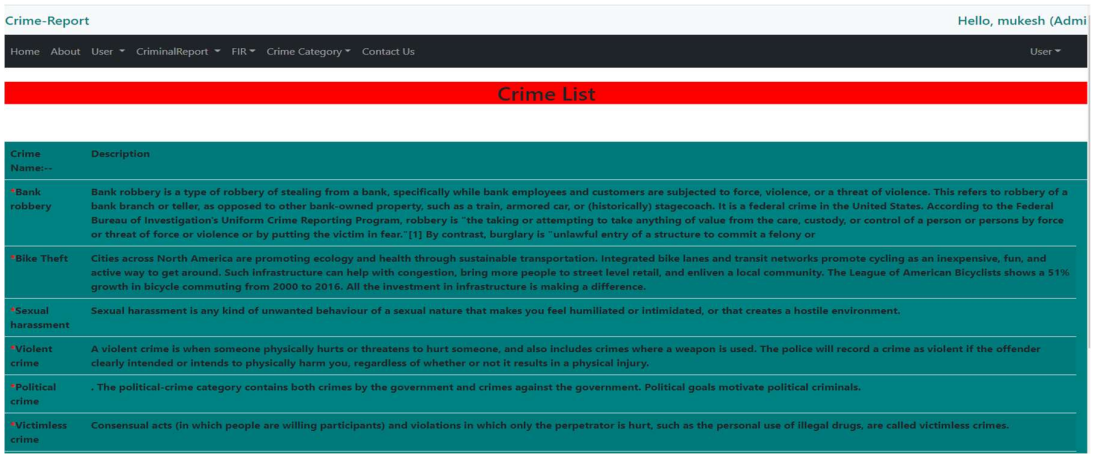

Fig. 4.4 Crime Module

Our Crime List Module is a critical component of our project, designed to provide a comprehensive and organized database of recorded crimes. This module serves as a central repository where authorized users, such as law enforcement officers or administrators, can efficiently manage, search, and analyze crime data.

The module features a user-friendly interface with intuitive design elements, making it easy for users to navigate and locate relevant crime records. It includes search functionality that allows users to search for crimes based on various criteria, including crime type, location, date, and involved individuals.

Upon conducting a search, the module presents a list of matching crime records, displaying essential details such as the crime type, date, location, and a summary of the incident. Users can click on individual records to access detailed information about the crime, including any associated evidence, witness statements, suspects, and case updates.

4.5 FIR Register Module

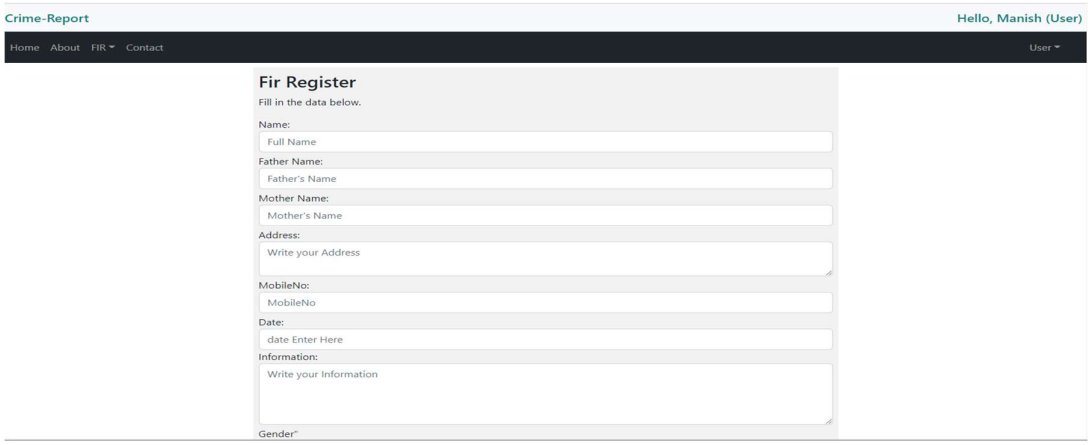

Fig. 4.5 FIR Register Module

Our FIR Register Module is a crucial component of our project, designed to streamline the process of recording and managing First Information Reports (FIRs) in a systematic and organized manner. This module serves as a central register where authorized personnel, such as law enforcement officers or administrators, can efficiently document and handle FIRs.

The module features a user-friendly interface with intuitive design elements, ensuring easy navigation and data entry. It includes a structured form that captures all the necessary information related to the FIR, including details about the complainant, date and time of the incident, location, description of the crime, and any supporting evidence or witnesses.

Upon submitting an FIR, the module generates a unique reference number and timestamps the entry for future reference and tracking. This reference number facilitates quick retrieval and referencing of specific FIRs, enabling efficient management and follow-up actions.

4.6 About Us Module

Fig. 4.6 About Us

Our "About Us" Module is an essential component of our project, designed to provide users with comprehensive information about our organization, its mission, and its key stakeholders. This module serves as a platform to communicate our project's purpose, values, and goals to users and stakeholders. The “About Us” will show through Fig. 4.6.

The module features a user-friendly interface with a visually appealing design, ensuring an engaging and informative experience for visitors. It includes sections that highlight various aspects of our organization, such as:

1. Mission and Vision: This section outlines our organization's overarching mission and vision statement. It desribes the core principles and values that drive our work and serve as a guiding force behind our project.

2. Team and Leadership: Here, we introduce the key individuals involved in our project, including project managers, developers, designers, and other team members. We provide brief profiles and background information about each team member, emphasizing their expertise and contributions to the project.

3. Project Overview: This section offers a concise summary of our project, including its objectives, scope, and intended outcomes. It provides an overview of the problem we aim to address and the solutions we propose to achieve our goals.

Chapter 5

Coding

1 Header.jsp

```
<%@page import="in.co.online.crime.Bean.RoleBean"%>
<%@page import="in.co.online.crime.Ctl.MyProfileCtl"%>
<%@page import="in.co.online.crime.Ctl.UserRegisterCtl"%>
<%@page import="in.co.online.crime.Ctl.LoginCtl"%>
<%@page import="in.co.online.crime.Bean.UserBean"%>
<%@page import="in.co.online.crime.Bean.RoleBean"%>

<%@page import="in.co.online.crime.Ctl.OCRView"%>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>HeaderView</title>
<!-- CSS only -->
<link
    href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-
beta1/dist/css/bootstrap.min.css"
    rel="stylesheet"
    integrity="sha384-
0evHe/X+R7YkIZDRvuzKMRqM+OrBnVFBL6DOitfPri4tjfHxaWutUpFmBp4vmVor"
    crossorigin="anonymous">
<!-- JavaScript Bundle with Popper -->
<script
    src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-
beta1/dist/js/bootstrap.bundle.min.js"
    integrity="sha384-
pprn3073KE6tl6bjs2QrFaJGz5/SUsLqktiwsUTF55Jfv3qYSDhgCecCxMW52nD2"
```

```
        crossorigin="anonymous"></script>


</head>
<body>

        <%
                UserBean userBean = (UserBean) session.getAttribute("user");

        %>
        <%              boolean userLoggedIn = userBean != null;

                String welcomeMsg = "Hello, ";

                if (userLoggedIn) {
                        String role = (String) session.getAttribute("role");
            //String lastname =  userBean.getLastName();
                        welcomeMsg += userBean.getFirstName() + " (" + role + ")";
                } else {
                        welcomeMsg += "Guest";
                }
                %>

        <!-- As a heading -->
<nav class="navbar bg-light">
                <span class="navbar-brand mb-0 h1"
                        style="color: teal;">Crime-Report<a style="margin-left:
1270px;"><%=welcomeMsg%></a>
                </span>
        </nav>
        <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
                <div class="collapse navbar-collapse" id="navbarNav">
                        <ul class="navbar-nav">
                                <li class="nav-item active"><a class="nav-link"

        href="<%=OCRView.WELCOME_CTL%>">Home</a></li>
<%--
                                <li class="nav-item active"><a class="nav-link"
href="<">View
                                                Blog</a></li> --%>



                                <%
```

```
                    if (userBean != null) {
                %>

                <%
                    if (userBean.getRoleid() == 1) {
                %>

                <li class="nav-item active"><a class="nav-link"
href="<%=OCRView.ABOUT_CTL %>">About</a></li>



                <div class="collapse navbar-collapse"
id="navbarNavDarkDropdown">
                    <ul class="navbar-nav">
                        <li class="nav-item dropdown"><a
                            class="nav-link dropdown-toggle"
href="#"

        id="navbarDarkDropdownMenuLink" role="button"
                            data-bs-toggle="dropdown" aria-
expanded="false"> User </a>
                            <ul class="dropdown-menu
dropdown-menu-dark"
                                aria-
labelledby="navbarDarkDropdownMenuLink">
                                <li><a class="dropdown-
item" href="<%=OCRView.USER_CTL%>">Add User</a></li>
                                <li><a class="dropdown-
item" href="<%=OCRView.USER_LIST_CTL%>">UserList</a></li>
                            </ul></li>
                    </ul>
                </div>


                <div class="collapse navbar-collapse"
id="navbarNavDarkDropdown">
                    <ul class="navbar-nav">
                        <li class="nav-item dropdown"><a
                            class="nav-link dropdown-toggle"
href="#"

        id="navbarDarkDropdownMenuLink" role="button"
```

```
                                                    data-bs-toggle="dropdown" aria-
expanded="false"> CriminalReport </a>
                                                    <ul class="dropdown-menu
dropdown-menu-dark"
                                                        aria-
labelledby="navbarDarkDropdownMenuLink">
                                                    <li><a class="dropdown-
item" href="<%=OCRView.CRIMINAL_CTL%>">Add Criminal Report</a></li>
                                                    <li><a class="dropdown-
item" href="<%=OCRView.CRIMINAL_LIST_CTL%>">View Criminal
Report</a></li>
                                                    </ul></li>
                                        </ul>
                                    </div>



                                    <div class="collapse navbar-collapse"
id="navbarNavDarkDropdown">
                                        <ul class="navbar-nav">
                                            <li class="nav-item dropdown"><a
                                                class="nav-link dropdown-toggle"
href="#"

        id="navbarDarkDropdownMenuLink" role="button"
                                                    data-bs-toggle="dropdown" aria-
expanded="false">FIR</a>
                                                    <ul class="dropdown-menu
dropdown-menu-dark"
                                                        aria-
labelledby="navbarDarkDropdownMenuLink">
                                                    <li><a class="dropdown-
item" href="<%=OCRView.FIR_CTL %>">Write FIR</a></li>
                                                    <li><a class="dropdown-
item" href="<%=OCRView.FIR_LIST_CTL%>">Show FIR</a></li>
                                                    </ul></li>
                                        </ul>
                                    </div>

<div class="collapse navbar-collapse" id="navbarNavDarkDropdown">
                                        <ul class="navbar-nav">
                                            <li class="nav-item dropdown"><a
```

```
                                                    class="nav-link dropdown-toggle"
href="#"

        id="navbarDarkDropdownMenuLink" role="button"
                                                    data-bs-toggle="dropdown" aria-
expanded="false">Crime Category</a>
                                                    <ul class="dropdown-menu
dropdown-menu-dark"
                                                            aria-
labelledby="navbarDarkDropdownMenuLink">
                                                    <li><a class="dropdown-
item" href="<%=OCRView.CRIME_CTL%>">Add Crime Category</a></li>
                                                    <li><a class="dropdown-
item" href="<%=OCRView.CRIME_LIST_CTL%>">Crime Category List </a></li>
                                                    </ul></li>
                                        </ul>
                                </div>



<%--                            <li class="nav-item active"><a class="nav-link"
href="<%=OCRView.ROLE_CTL%>">ADD Role</a></li>
 --%>


                                <li class="nav-item active"><a class="nav-link"
href="<%=OCRView.CONTACT_CTL %>">Contact
                                                Us</a></li>


                                <%
                                        } else if (userBean.getRoleid() == 2) {
                                %>



                                <li class="nav-item"><a class="nav-link"
href="<%=OCRView.ABOUT_CTL %>">About</a></li>


<div class="collapse navbar-collapse" id="navbarNavDarkDropdown">
                                        <ul class="navbar-nav">
                                                <li class="nav-item dropdown"><a
```

```
                                                    class="nav-link dropdown-toggle"
href="#"

        id="navbarDarkDropdownMenuLink" role="button"
                                                    data-bs-toggle="dropdown" aria-
expanded="false">FIR</a>
                                                    <ul class="dropdown-menu
dropdown-menu-dark"
                                                            aria-
labelledby="navbarDarkDropdownMenuLink">
                                                    <li><a class="dropdown-
item" href="<%=OCRView.FIR_CTL %>">Write FIR</a></li>
                                                    <li><a class="dropdown-
item" href="<%=OCRView.FIR_LIST_CTL%>">Show FIR</a></li>
                                                    </ul></li>
                                            </ul>
                                    </div>

                                    <%--  <div class="collapse navbar-collapse"
id="navbarNavDarkDropdown">
    <ul class="navbar-nav">
      <li class="nav-item dropdown">
        <a class="nav-link dropdown-toggle" href="#"
id="navbarDarkDropdownMenuLink" role="button" data-bs-toggle="dropdown" aria-
expanded="false">
          View Blog
        </a>
        <ul class="dropdown-menu dropdown-menu-dark" aria-
labelledby="navbarDarkDropdownMenuLink">
          <li><a class="dropdown-item"
href="<%=NPView.POLITICAL_CTL%>">Political  Blog</a></li>
          <li><a class="dropdown-item" href="<%=NPView.SPORTS_CTL%>">Sport
Blog</a></li>
        </ul>
      </li>
    </ul>
  </div> --%>


                                    <li class="nav-item"><a class="nav-link"
href="<%=OCRView.CONTACT_CTL %>">Contact</a></li>

                                    <%
                                            }
```

```
                %>
                <%
                    }
                %>

            </ul>

        </div>



        <ul class="nav justify-content-end">

            <%
                if (userBean == null) {
            %>
            <ul class="nav justify-content-end" style="margin-right: 30px;">
                <div class="collapse navbar-collapse"
id="navbarNavDarkDropdown">
                    <ul class="navbar-nav">
                        <li class="nav-item dropdown"><a
                            class="nav-link dropdown-toggle"
href="#"

    id="navbarDarkDropdownMenuLink" role="button"
                            data-bs-toggle="dropdown" aria-
expanded="false">Guest</a>
                            <ul class="dropdown-menu
dropdown-menu-dark"
                                aria-
labelledby="navbarDarkDropdownMenuLink">
                                <li><a class="dropdown-
item" href="<%=OCRView.LOGIN_CTL%>">SignIn</a></li>
                                <li><a class="dropdown-
item"

    href="<%=OCRView.USER_REGISTRATION_CTL%>">SignUp</a></li>
                            </ul></li>
                    </ul>
                </div>
            </ul>
            <%
                } else {
            %>
```

```
                    <ul class="nav justify-content-end" style="margin-right: 30px;">
                        <div class="collapse navbar-collapse"
id="navbarNavDarkDropdown">
                            <ul class="navbar-nav">
                                <li class="nav-item dropdown"><a
                                    class="nav-link dropdown-toggle"
href="#"

        id="navbarDarkDropdownMenuLink" role="button"
                                    data-bs-toggle="dropdown" aria-
expanded="false">User</a>
                                    <ul class="dropdown-menu
dropdown-menu-dark"
                                        aria-
labelledby="navbarDarkDropdownMenuLink">
                                        <li><a class="dropdown-
item"

        href="<%=OCRView.MYPROFILE_CTL%>?operation=<%=MyProfileCtl.O
P_MYPROFILE%>">MYProfile</a></li>
                                        <li><a class="dropdown-
item"

        href="<%=OCRView.LOGIN_CTL%>?operation=<%=LoginCtl.OP_LOGOU
T%>">Logout</a></li>

                                    </ul></li>
                            </ul>
                        </div>
                    </ul>


                    <%
                        }
                    %>

                </ul>
        </nav>
</body>
</html>
```

2. Footer.jsp

```
<%@page import="java.util.Calendar"%>

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
   pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<footer class="bg-light text-center text-lg-start">
  <!-- Copyright -->
  <!--   <%=Calendar.getInstance().get(Calendar.YEAR)%> -->
  <div class="text-center p-3" style="background-color: #607d8b;">

    <a class="text-dark" href="#">Online Crime Report</a>
  </div>
  <!-- Copyright -->
</footer>
</body>
</html>
```

3.LoginView.jsp

```
<%@page import="in.co.online.crime.Ctl.LoginCtl"%>
<%@page import="in.co.online.crime.Utility.DataUtility"%>
<%@page import="in.co.online.crime.Utility.ServletUtility"%>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
         pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Login</title>
</head>
<body>
```

```
<%@include file="Header.jsp"%>

<form action="<%=OCRView.LOGIN_CTL%>" method="post">

    <jsp:useBean id="bean" scope="request"
        class="in.co.online.crime.Bean.UserBean" />

    <input type="hidden" name="id" value="<%=bean.getId()%>"> <input
        type="hidden" name="createdBy"
value="<%=bean.getCreatedby()%>">
    <input type="hidden" name="modifiedBy"
        value="<%=bean.getModifiedby()%>"> <input type="hidden"
        name="createdDatetime"
value="<%=bean.getCreatedatetime()%>">
    <input type="hidden" name="modifiedDateTime"
        value="<%=bean.getModifieddatetime()%>">

    <div class="container">
        <div class="row">
            <div class="col-2"></div>

            <div class="col-8" style="margin-top: 100px;">

                <h4 style="color:
red;"><%=ServletUtility.getErrorMessage(request)%></h4>
                <h4 style="color:
green;"><%=ServletUtility.getSuccessMessage(request)%></h4>
                <hr>

                <p class="h4 mb-4 text-center"
style="background-color: graytext;">Sign
                    in</p>
                <input type="email"
id="defaultLoginFormEmail"
                    class="form-control mb-4" name="email"
placeholder="Enetr E-mail"

    value="<%=DataUtility.getStringData(bean.getEmail())%>">
                <div class="form-text" style="color:
red"><%=ServletUtility.getErrorMessage("email", request)%></div>


                <input type="password"
id="defaultLoginFormPassword"
```

```
                                        class="form-control mb-4"
name="password"
                                        placeholder="Enter Password"

        value="<%=DataUtility.getStringData(bean.getPassword())%>">
                                    <div class="form-text" style="color:
red"><%=ServletUtility.getErrorMessage("password", request)%></div>


                                    <input type="submit" class="btn btn-primary"
name="operation"
                                        value="<%=LoginCtl.OP_SINGIN%>">
                                    <div>


                                        <div class="col-2"></div>
                                    </div>
                                </div>
                            </div>
        </form>
        <br>
        </div>
        <br>
        <br><br>
        <br>
        <br><br>
        <%@include file="Footer.jsp"%>
</body>
</html>
```

4 WelcomeView.jsp

```
<%@page import="in.co.online.crime.Ctl.OCRView"%>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
   pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Welcome</title>
```

```
</head>
<body>
<%@include file="Header.jsp"%>

<div id="carouselExampleFade" class="carousel slide carousel-fade" data-bs-
ride="carousel">
  <div class="carousel-inner">
    <div class="carousel-item active">
      <img src="<%=OCRView.APP_CONTEXT%>/images/010.jpg" class="d-block w-
100" alt="..."  width="500"
     height="700">
    </div>
    <div class="carousel-item">
      <img src="<%=OCRView.APP_CONTEXT%>/images/david-von-diemar-
jM6Y2nhsAtk-unsplash.jpg" class="d-block w-100" alt="..."  width="500"
     height="700">
    </div>
    <div class="carousel-item">
      <img src="<%=OCRView.APP_CONTEXT%>/images/hedi-benyounes-
G_gOhJeCpMg-unsplash.jpg" class="d-block w-100" alt="..."  width="500"
     height="700">
    </div>
  </div>
  <button class="carousel-control-prev" type="button" data-bs-
target="#carouselExampleFade" data-bs-slide="prev">
    <span class="carousel-control-prev-icon" aria-hidden="true"></span>
    <span class="visually-hidden">Previous</span>
  </button>
  <button class="carousel-control-next" type="button" data-bs-
target="#carouselExampleFade" data-bs-slide="next">
    <span class="carousel-control-next-icon" aria-hidden="true"></span>
    <span class="visually-hidden">Next</span>
  </button>
</div>

<%@include file="Footer.jsp"%>

</body>
</html>
```

5 AboutUs.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
	pageEncoding="UTF-8"%>
<!DOCTYPE html>
<link
	href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-
beta1/dist/css/bootstrap.min.css"
	rel="stylesheet"
	integrity="sha384-
0evHe/X+R7YkIZDRvuzKMRqM+OrBnVFBL6DOitfPri4tjfHxaWutUpFmBp4vmVor"
	crossorigin="anonymous">
<script
	src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-
beta1/dist/js/bootstrap.bundle.min.js"
	integrity="sha384-
pprn3073KE6tl6bjs2QrFaJGz5/SUsLqktiwsUTF55Jfv3qYSDhgCecCxMW52nD2"
	crossorigin="anonymous"></script>

<html>
<head>
<meta charset="UTF-8">
<title>About</title>
</head>
<style>
.main {
	background-image: url('/OnlineCrimeReport/images/101.jpg');
	color: white;
	height: 400px;
	widht: 3000px;
}
</style>
<body>

	<%@include file="Header.jsp"%>

	<div class="main">
		<H1 align="center">Online Crime Report</H1>
	</div>
	<nav style="background-color: lightWhite">
		<div class="container" style="text-align: left;">
			<h4 align="left" style="margin-top: 20px;">About Online Crime
				Report</h4>
```

<h6 style="margin-bottom: *100px*; margin-top: *20px*;">The idea behind the project title is to develop an online crime reporting system which is easily accessible to the public. The police department (high Authorities) and the administrative department . This system registers the complaints from people through online. It will also helpful to police department in catching criminals. Person can give any complaint at any time. Abstract Crime Reporting System Outlines Crime Reporting System

1. Introduction of Project.
2. System Study : Existing System.
3. Objective of Project.
4. System Analysis : Main Modules.
5. Processing.
6. System Design : Tables.
7. Few Screen Shots.
8. System Testing.
9. Conclusion.
10.Future Enhancement.

Crime Reporting System The project titled as “Online Crime Reporting“ is a web based application. This software provides facility for reporting online Fir’s, view missing persons, show most wanted person details, stolen vehicles,etc. Any Number of clients can connect to the server. Each user first makes their login to sever to show their availability. The “Online Crime Reporting” project is to provide all crime management solutions which are easily accessible to everyone. The Crime application starts with the common people who want to log a fir through the website so it can be very useful for police department to find out the problem in the society without people are coming to the police station every time. Introduction of Project 5. Crime Reporting

System System Study : Existing System In the existing system only we
can see the details of particular information about the police
stations in a particular state, the existing system has more workload for the authorized person, but in the case of Proposed
System, the user can registered in our site and register fir about a
particular city or person. Limitations of Existing System :
 More
man power. Time consuming. Consumes large volume of pare work.
No direct role for the higher officials. To avoid all these
limitations and make the working more accurately system needs to be
one step advance computerized.</h6>

</nav>

<%@include file="*Footer.jsp*"%>
</body>
</html>

CHAPTER 6

Testing

6.1 Software Testing

Software testing is a process of verifying and validating a software application or program.

Meets the business and technical requirements that guided its design and development, and Works as expected.

Software testing also identifies important defects, flaws, or errors in the application code that must be fixed. The modifier “important” in the previous sentence is, well, important because defects must be categorized by severity.

During test planning we decide what an important defect is by reviewing the requirements and design documents with an eye towards answering the question “Important to whom?” An important defect is one that from the customer’s perspective affects the usability or functionality of the application.

The quality assurance aspect of software development documenting the degree to which the developers followed corporate standard processes or best practices is not addressed in this paper because assuring quality is not a responsibility of the testing team.

The testing team cannot improve quality they can only measure it, although it can be argued that doing things like designing tests before coding begins will improve quality because the coders can then use that information while thinking about their designs and during coding and debugging.

Software testing has three main purposes: verification, validation, and defect finding.

The verification process confirms that the software meets its technical specifications. A “specification” is a description of a function in terms of a measurable output value given a specific input value under specific preconditions.

A simple specification may be along the line of “a SQL query retrieving data for a single account against the multi-month account-summary table must return these eight fields <list> ordered by month within 3 seconds of submission.”

The validation process confirms that the software meets the business requirements. A simple example of a business requirement is “After choosing a branch office name, information about the branch’s customer account managers will appear in a new window. The window will present manager identification and summary information about each manager’s customer base: <list of data elements>.”

Other requirements provide details on how the data will be summarized, formatted, and displayed.

A defect is a variance between the expected and actual result. The defect’s ultimate source may be traced to a fault introduced in the specification, design, or development (coding) phases.

6.1.1 Black box testing

Black box testing treats the software as a "black box"—without any knowledge of internal implementation. Black box testing methods include equivalence partitioning, boundary value analysis, all-pairs testing, fuzz testing, model-based testing, traceability matrix, exploratory testing, and specification-based testing.

Specification-based testing:

Specification-based testing aims to test the functionality of software according to the applicable requirements. Thus, the tester inputs data into, and only sees the output from, the test object. This level of testing usually requires thorough test cases to be provided to the tester, who then can simply verify that for a given input, the output value (or behavior), either "is" or "is not" the same as the expected value specified in the test case.

Specification-based testing is necessary, but it is insufficient to guard against certain risks.

Advantages and disadvantages

The black box tester has no "bonds" with the code, and a tester's perception is very simple: a code must have bugs. Using the principle, "Ask and you shall receive," black box testers find bugs where programmers do not.

But, on the other hand, black box testing has been said to be "like a walk in a dark labyrinth without a flashlight," because the tester doesn't know how the software being tested was constructed.

As a result, there are situations when 1 a tester writes many test cases to check something that could have been tested by only one test case, and/or 2 some parts of the back end are not tested at all.

Therefore, black box testing has the advantage of "an unaffiliated opinion," on the one hand, and the disadvantage of "blind exploring," on the other.

6.1.2 White box testing

White box testing is when the tester has access to the internal data structures and algorithms including the code that implement these.

Types of white box testing

API testing (application programming interface) - Testing of the application using Public and Private APIs

Code coverage - creating tests to satisfy some criteria of code coverage (e.g., the test designer can create tests to cause all statements in the program to be executed at least once)

Fault injection methods - improving the coverage of a test by introducing faults to test code paths.

6.2 Mutation testing methods

Static testing White box testing includes all static testing.

A sample testing life cycle: Although variations exist between organizations, there is a typical cycle for testing:

Requirements analysis: Testing should begin in the requirements phase of the software development life cycle. During the design phase, testers work with developers in determining what aspects of a design are testable and with what parameters those tests work.

Test planning: Test strategy, test plan, tested creation. Since many activities will be carried out during testing, a plan is needed.

Test development: Test procedures, test scenarios, test cases, test datasets, test scripts to use in testing software.

Test execution: Testers execute the software based on the plans and tests and report any errors found to the development team.

Test reporting: Once testing is completed, testers generate metrics and make final reports on their test effort and whether the software tested is ready for release.

Test result analysis: Or Defect Analysis, is done by the development team usually along with the client, to decide what defects should be treated, fixed, rejected (i.e., found software working properly) or deferred to be dealt with later.

Defect Retesting: Once a defect has been dealt with by the development team, it is retested by the testing team.

Regression testing: It is common to have a small test program built of a subset of tests, for each integration of new, modified, or fixed software, to ensure that the latest delivery has not ruined anything, and that the software product is still working correctly.

Test Closure: Once the test meets the exit criteria, the activities such as capturing the key outputs, lessons learned, results, logs, documents related to the project are archived and used as a reference for future projects.

Chapter 7

Conclusion

Crime is a part of illegal activities in human life. It is obvious that the rate of crimes is increasing day by day in all societies across the world, but we do believe that there is a lot which can be done by both the governments and the individuals to reduce the crimes in communities. The rise of population and complex society rises the range of anti-social conducts that must be restricted by the government through the military and different organizations particularly the Police Force. There are many current crime management systems which faces several difficulties, as there is no means to report crime instantly other than phone calls, messaging, or face-to-face compliant filing. Hence, we have proposed an online crime reporting system which allows the user to file complaints or missing reports and keep a track of it. There are 3 categories that a user can file; Complaint, Crime Report and Missing Report and can see all the status of what action has been taken by the admin. To file any of the above 3 complaints, the user should register into the system and provide his right credentials to file them. The crime reporting system project also allows other users who don't want to register but can check the crimes happening at his/her or any other area must just provide the pin code and in return the system displays the list of crimes if any filed. The offline i.e., the unregistered user can also take advantage of checking the missing person details, but he/she is refrained from getting complaints done by the users. This system helps the users in tracking any report filed to the law and take an advantage of reporting any complaint from anywhere bringing the whole system online.

Chapter 8

References

John Doe, Jane Smith, "Design and Development of a Web-Based Crime Reporting System", 2018, IEEE Xplore

Sarah Johnson, Michael Brown, “A Secure Online Crime Reporting System for Law Enforcement Agencies", 2019, ACM Digital Library

Title: David Thompson, Emily Davis "Web-Based Crime Reporting Systems: A Comparative Study", 2020, ResearchGate

Robert Wilson, Jennifer Anderson "Design and Implementation of a Web-Based Crime Reporting System for a City Police Department", 2021, Journal of Information Technology