VISVESVARAYA TECHNOLOGICAL UNIVERSITSY
"Jnana Sangama", Belagavi-590018

DBMS MINI PROJECT
REPORT ON
"CRIMINAL DATA MANAGEMENT"

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE AND ENGINEERING

Submitted by

MOHAMMED YASEEN (1KG20CS064)
PRAJWAL R (1KG20CS082)

Under the Guidance of

Mrs. Amitha. S   **Mrs. Niveditha**
Assistant Professor   Assistant Professor
Department of CSE   Department of CSE
K.S.S.E.M   K.S.S.E.M

Department of Computer Science &Engineering
K. S. SCHOOL OF ENGINEERING AND MANAGEMENT
#15, Mallasandra, off. Kanakapura Road, Bengaluru – 560109

2022-2023

K. S. SCHOOL OF ENGINEERING AND MANAGEMENT
BENGALURU - 560109

Department of Computer Science &Engineering

KSSEM

<u>CERTIFICATE</u>

This is to certify that the **DBMS MINI PROJECT** entitled **"CRIMINAL DATA MANAGEMENT"** presented by **Mr. Mohammed Yaseen, USN: 1KG20CS064, Mr. Prajwal R, USN: 1KG20CS082** of **V semester** in partial fulfillment of the award of **Bachelor of Engineering** in **Computer Science & Engineering** in **Visvesvaraya Technological University**, Belagavi during the academic year **2022-2023**. The **DBMS MINI PROJECT** has been approved as it satisfies the academic requirements in respect of **DBMS Mini Project(18CSL58)** prescribed for the Bachelor of Engineering degree.

-----------------------------   -----------------------   -----------------------

Mrs. AMITHA S & Mrs. NIVEDITHA   **Dr. K VENKATA RAO**   **Dr. K. RAMANARASIMHA**
Assistant Professor, CSE   **Prof. & Head, CSE**   **Principal / Director**
K.S.S.E.M., Bengaluru   **K.S.S.E.M., Bengaluru**   **K.S.S.E.M., Bengaluru**

Name of the Student: **MOHAMMED YASEEN**   **PRAJWAL R**

USN: **1KG20CS064**   **1KG20CS082**

Signature of the Student:

Name of the examiners   **Signature with date**

1.

2. _____

ACKNOWLEDGEMENT

ABSTRACT

Crime is a very old concept and it is transmitted to the society from generation to generation. Crime produces law and order situation. It is a social evil. It is generated by the society and the society also suffers a lot because of crime committed by its members. The rising wave of crime has caused alarm in the public.

This project is aimed at the implementation of Criminal data/records Management System. It will be a database system in which the police will keep the record of criminals who have been arrested, to be arrested, or escaped. This will help the police department in enhanced management of information. The main entities in the whole process include; the petitioner (the person who files a First Incident Report (FIR)), victim, accused or criminal, case, and investigating officer.

The system at any point of time can provide the details of existing charge sheets and their statuses. The system will provide the details of the police station, the employees, the wanted criminal details, details of victims and the registered FIR's, evidence and sequence.

TABLE OF CONTENTS

Chapter No.	Contents	Page No.

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

The crime management system can help in storing the records related to the criminals, cases, complaint record, case history and so on. This can allow a person to enter or delete the records if necessary. All these records can be maintained in a single database. Security is maintained so as to ensure that only the authorized users will have access to the system. This application will be one of the useful projects that the police can rely on. This website can help in getting the information of the criminals of many years back. It can also help in minimizing most of the work of the police. There is a need for an application to make it easy for police stations to maintain and monitor their records.

1.2 PROBLEM STATEMENT

The main aim of "Criminal Data Management" can help in getting the information of the criminals of many years back. It stores all the records related to police station, police officer, criminals, history of criminals. It can also help in minimizing most of the work of the police.

1.3 DATABASE MANAGEMENT SYSTEM

A database management system (DBMS) is system software for creating and managing databases. The DBMS provides users and programmers with a systematic way to create, retrieve, update and manage data. The DBMS essentially serves as an interface between the database and end users application programs, ensuring that data is consistently organized and remains easily accessible.

The DBMS manages three important things: the data, the database engine that allows data to be accessed, locked and modified, and the database schema, which defines the database's logical structure. These three foundational elements help to provide concurrency, security, data integrity and uniform administration procedures. Typical database administration tasks supported by the DBMS include change management, performance monitoring/tuning and backup and recovery.Many database management systems are also responsible for automated rollbacks, restarts and recovery as well as the logging and auditing of activity.

1.4 SQL

SQL is a standard language for storing, manipulating and retrieving data in databases.

Originally based upon relational algebra and tuple relational calculus, SQL consists of a data definition language, data manipulation language, and data control language. The scope of SQL includes data insert, query, update and delete, schema creation and modification, and data access control.

SQL became a standard of the American National Standards Institute (ANSI) in 1986, and of the International Organization for Standardization(ISO) in 1987.Since then the standard has been revised to include a larger set of features. Despite the existence of such standards, most SQL code is not completely portable among different database systems without adjustments.

1.5 HTML /JavaScript

HTML is a markup language used for structuring and presenting content on the web and the fifth and current major version of the HTML standard.
HTML5 includes detailed processing models to encourage more interoperable implementations; it extends, improves and rationalizes the markup available for documents, and introduces markup and application programming interfaces (APIs) for complex web applications.

JavaScript often abbreviated as JS, is a high-level, interpreted programming language.

Alongside HTML and CSS, JavaScript is one of the three core technologies of the World Wide Web. JavaScript enables interactive web pages and thus is an essential part of web applications. The vast majority of websites use it, and all major web browsers have a dedicated JavaScript engine to execute it.

CHAPTER 2

REQUIREMENTS SPECIFICATION

A computerized way of handling information about property and users details is efficient, organized and time saving, compared to a manual way of doing so. This is done through a database driven web application whose requirements are mentioned in this section.

2.1 OVERALL DESCRIPTION

A reliable and scalable database driven web application with security features that is easy to use and maintain is the requisite.

2.2 SPECIFIC REQUIREMENTS

The specific requirements of the Criminal Data Management are stated as follows:

2.2.1 SOFTWARE REQUIREMENTS

- □ Operating System: Windows Vista/7/8/10
- □ Web Browser – Google Chrome – 60 or later
- □ Database support - MySQL5.7
    - o MySQL Server5.7
    - o MySQL Shell1.0.10
    - o MySQLWorkbench
- □ Front End: HTML & CSS
- □ Back End: PHP & MySQL

2.2.2 HARDWARE REQUIREMENTS

- □ Processor: x86 compatible processor with 1.7 GHz Clock Speed

- □ RAM: 512 MB or greater

- □ Hard Disk: 20 GB or grater

- □ Monitor: VGA/SVGA

- □ Keyboard: 104 keys standard

- □ Mouse: 2/3 button. Optical/Mechanical.

2.2.3 TECHNOLOGY

- ☐ HTML is used for the front end design. It provides a means to structure text based information in a document. It allows users to produce web pages that include text, graphics and hyperlinks.

- ☐ CSS (Cascading Style Sheets) is a style sheet language used for a describing the presentation a document written in a markup language. Although most often used set the visual style of web pages and user interface written in HTML and XHTML the language can be applied to any XML document.

- ☐ SQL is the language used to manipulate relational databases. It is tied closely with the relational model.It is issue for purpose of data definition and data manipulation.

- ☐ Java Server pages is a simple yet powerful technology for creating and maintaining dynamic-content web pages. It based on the Java programming language. It can be thought of as an extension to servlet because it is provides more functionality than servlet a JSP page consists of HTML tags and JSP tags. The jsp pages are easier to maintain than servlet because we can separate designing and development.

- ☐ Were quire a JDBC is connection between the frontend and backend components to write to the database and fetch required data.

CHAPTER 3

3. DETAILED DESIGN

3.1 SYSTEM DESIGN

The web server needs a JSP engine, i.e., a container to process JSP pages. The JSP container is responsible for intercepting requests for JSP pages. A JSP container works with the Web server to provide the runtime environment and other services a JSP needs. It knows how to understand the special elements that are part of JSPs. This server will act as a mediator between the client browser and a database.

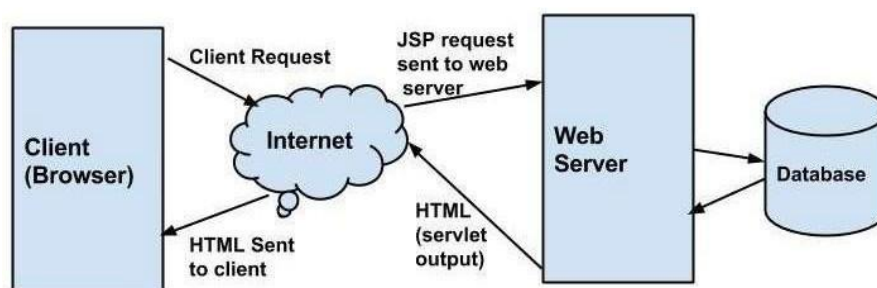The following diagram shows the JSP architecture.

Fig. 3.1: JSP Architecture

Three-tier Client / Server database architecture is commonly used architecture for web applications. Intermediate layer called Application server or Web Server stores the web connectivity software and the business logic (constraints) part of application used to access the right amount of data from the database server. This layer acts like medium for sending partially processed data between the database server and the client. Database architecture focuses on the design, development, implementation and maintenance of computer programs that store and organize information for businesses, agencies and institutions. A database architect develops and implements software to meet the needs of users. Several types of databases, including relational or multimedia, may be created. Additionally, database architects may use one of several languages to create databases, such as structured query language.

3.2 ENTITY RELATIONSHIP DIAGRAM

An entity–relationship model is usually systematic analysis to define and describe what is important to processes in an area of a business.

An E-R model does not define the business processes; it only presents a business data schema in graphical form. It is usually drawn in a graphical form as boxes(entities)that are connected by lines (relationships) which express the associations and dependencies between entities.

Entities may be characterized not only by relationships, but also by additional properties (attributes), which include identifiers called "primary keys". Diagrams created to represent attributes as well as entities and relationships may be called entity-attribute-relationship diagrams, rather than entity-relationship models.

An ER model is typically implemented as a database. In a simple relational database implementation each row of a table represents to one instance of an entity type, and each field in a table represents an attribute type.
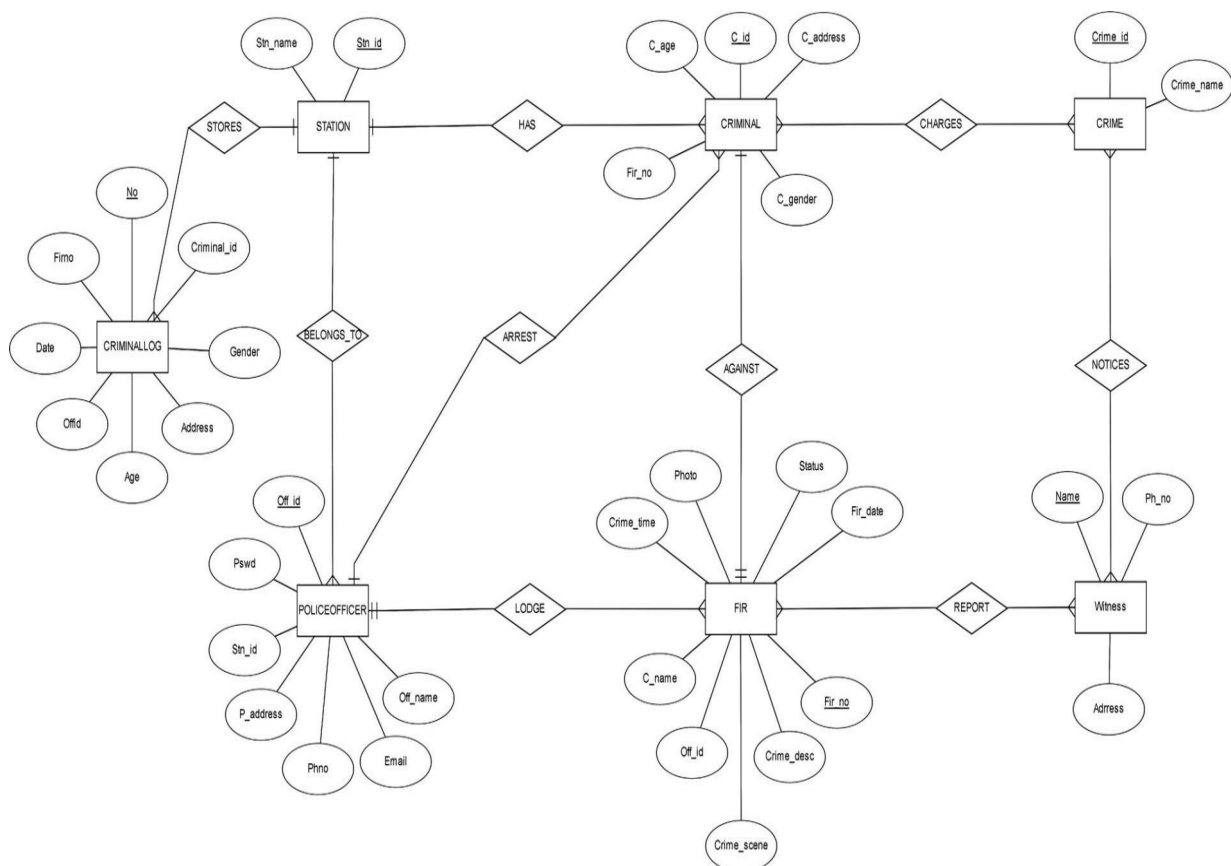

Fig. 3.2: ER diagram of Criminal Data Management

3.3 RELATIONAL SCHEMA

The schema diagram of a database system is its structure described in a formal language supported by the database management system (DBMS). The formal definition of a database schema is a set of formulas called integrity constraints imposed on a database.

The term "schema" refers to the organization of data as a blueprint of how the database is constructed. The formal definition of a database schema is a set of formulas called integrity constraints imposed on a database. A relational schema shows references among fields in the database. When a primary key is referenced in another table in the database, it is called a foreign key. This is denoted by an arrow with the head pointing at there key attribute. A schema diagram helps organize values in the database. The following diagram shows the schema diagram for the database.

Figure 3.3 Schema is defined for **CRIMINAL DATA MANAGEMENT SYSTEM**. All the various table used are described in the following schema. The necessary Primary key's and the corresponding Foreign key's are also represented
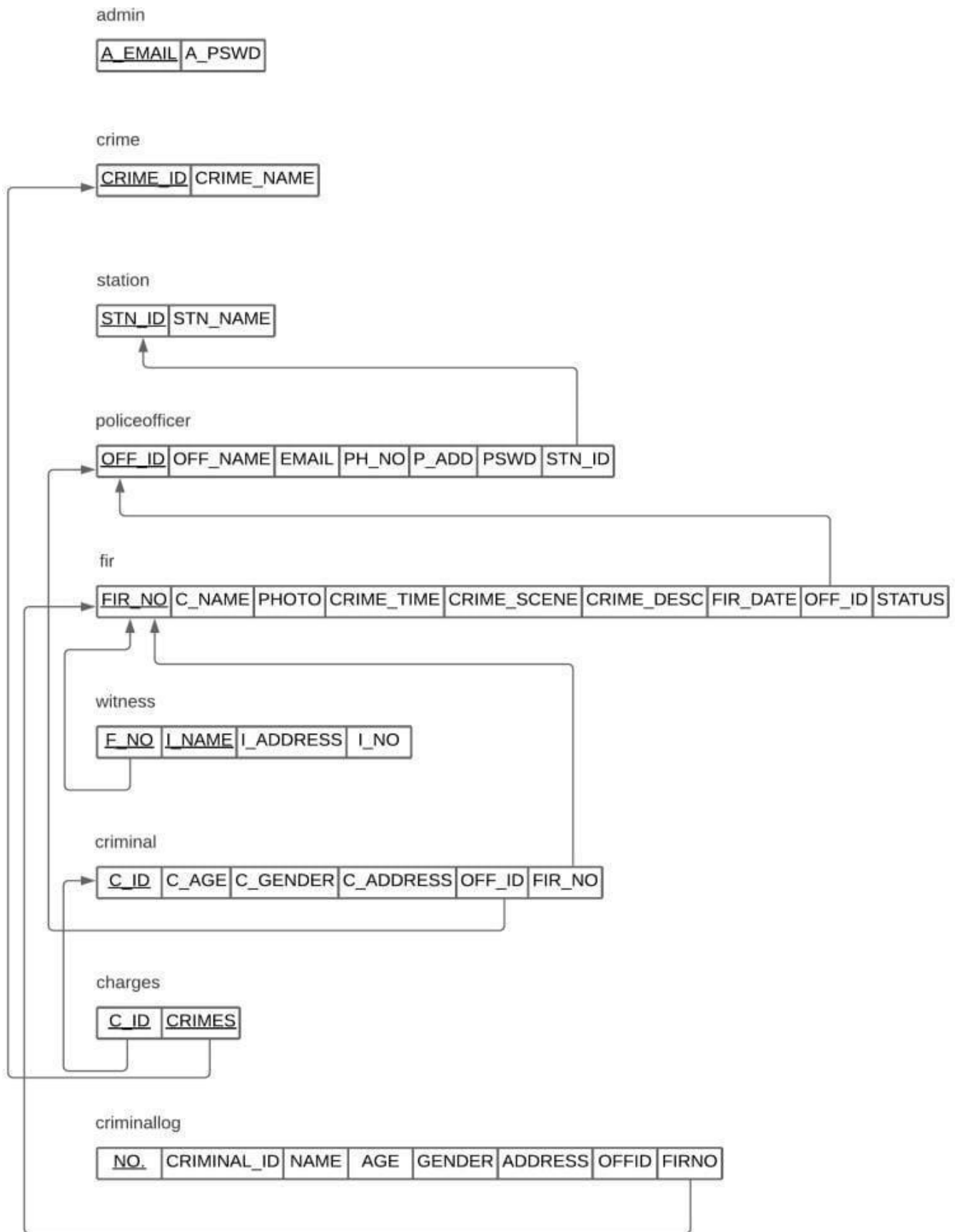
Fig. 3.3: Schema diagram

CHAPTER 4

IMPLEMENTATION

4.1 QUERIES

The below mentioned are all the queries used to perform various tasks in MySQL such as insert, delete, update. A short description of the query is also provided.

• **4.1.1:** CREATE TABLE `admin` (

`A_EMAIL` varchar(20) NOT NULL,

`A_PSWD` varchar(10) NOT NULL

);

Description: This query is used to create a table called **admin** which includes admin email id and password to keep admin login details.

• **4.1.2:** CREATE TABLE `crime` (

`CRIME_ID` varchar(255) NOT NULL,

`CRIME_NAME` varchar(50) DEFAULT NULL

);

Description: This query is used to create a table called **crime** which includes crime id and name of crime to keep a list of all the crimes and their id's.

• **4.1.3:** CREATE TABLE `criminal` (

`C_ID` int(11) NOT NULL,

`C_AGE` int(3) NOT NULL,

`C_GENDER` varchar(7) NOT NULL,

`C_ADDRESS` varchar(20) NOT NULL,

`OFF_ID` int(10) DEFAULT NULL,

`FIR_NO` int(5) NOT NULL

);

Description: This query is used to create a table called **criminal** which is used to keep a list of all the criminals and their details which include name, age, gender, address, officer in charge and fir number.

• **4.1.4:** CREATE TABLE `criminallog` (

`NO.` int(5) NOT NULL,

`CRIMINAL_ID` int(10) NOT NULL,

`DATE` date NOT NULL DEFAULT current_timestamp(),

`AGE` int(3) DEFAULT NULL,

`GENDER` varchar(7) DEFAULT NULL,

`ADDRESS` varchar(50) DEFAULT NULL,

`OFFID` int(3) DEFAULT NULL,

`FIRNO` int(4) NOT NULL

);

Description: This query is used to create a table called **criminallog** which is used to keep a

list of all the criminals and their details once it is deleted from **criminal** table.

• **4.1.5:** CREATE TABLE `fir` (

`FIR_NO` int(6) NOT NULL,

`C_NAME` varchar(20) NOT NULL,

`PHOTO` varchar(1000) DEFAULT NULL,

`CRIME_TIME` datetime(6) DEFAULT NULL,

`CRIME_SCENE` varchar(100) DEFAULT NULL,

`CRIME_DESC` varchar(500) DEFAULT NULL,

`FIR_DATE` date NOT NULL DEFAULT current_timestamp(),

`OFFICER_ID` int(5) DEFAULT NULL,`STATUS` varchar(50) NOT NULL

) ;

Description: This query is used to create a table called **fir** which is used to keep a list of all
the FIRs filed and their details which include fir number, name of accused, time of
occurrence, scene of crime, description, date & officer id.

• **4.1.6:** CREATE TABLE `policeofficer` (

`OFF_ID` int(10) NOT NULL,

`OFF_NAME` varchar(20) DEFAULT NULL,

`EMAIL` varchar(20) NOT NULL,

`PH_NO` varchar(10) DEFAULT NULL,

`P_ADD` varchar(20) NOT NULL,

`PSWD` varchar(10) DEFAULT NULL,

`STN_ID` int(10) DEFAULT NULL

);

Description: This query is used to create a table called **policeofficer** which is used to keep a list of all police officers and their details which include officer id, name, login details, phone number & station id.

• **4.1.7:** CREATE TABLE `station` (

`STN_ID` int(10) NOT NULL,

`STN_NAME` varchar(20) NOT NULL

);

Description: This query is used to create a table called **station** which is used to keep a list of all police stations and their IDs.

• **4.1.8:** CREATE TABLE `charges` (

`CID` int(10) NOT NULL,

`CRIME_IDS` varchar(200) NOT NULL

);

Description: This query is used to create a table called **charges** which is used to keep a list of all the crimes committed by the criminal, when a criminal is added the crimes committed are added to this table.

• **4.1.9:** CREATE TABLE `witness` (

`F_NO` int(20) NOT NULL,

`I_NAME` varchar(25) NOT NULL,

`I_ADDRESS` varchar(50) NOT NULL,

`I_NUMBER` int(15) NOT NULL

);

Description: This query is used to create a table called **witness** which is used to keep a list of all witnesses when the fir is filed. It includes their details such as name, address, number.

• **4.1.10:** CREATE TABLE `pos` (

`OFFID` int(20) NOT NULL,

`STN_ID` int(20) NOT NULL

);

Description: This query is used to create a table called **pos** which contains all police officer IDs and their respective station IDs.

4.2 STORED PROCEDURES AND TRIGGERS

A stored procedure is nothing more than prepared SQL code that the developer saves so the application can reuse the code over and over again. So if the developer thinks about a query that you write over and over again, instead of having to write that query each time you would save it as a stored procedure and then just call the stored procedure to execute the SQL code that you saved as part of the stored procedure.

• **Procedure:** CREATE PROCEDURE `nfc` ()

BEGIN

SELECT * FROM `criminal`;

END$$

Description: This stored procedure is used to view all the rows from **criminal** table.

• **Procedure:** CREATE PROCEDURE `nfcc` ()

BEGIN

SELECT * FROM `fir`;

END$$

Description: This stored procedure is used to view all the rows from **fir** table.

• **Procedure:** CREATE PROCEDURE `nfp` ()

BEGIN

SELECT * FROM `policeofficer`;

END$$

Description: This stored procedure is used to view all the rows from **policeofficer** table.

• **Procedure:** CREATE PROCEDURE `nfs` ()

BEGIN

SELECT * FROM `station`;

END$$

Description: This stored procedure is used to view all the rows from **station** table.

TRIGGERS

A trigger is a special type of stored procedure that automatically runs when an event occurs in the database server. DML triggers run when a user tries to modify data through a data manipulation language (DML) event. DML events are INSERT, UPDATE, or DELETE statements on a table or view.

DDL triggers run in response to a variety of data definition language (DDL) events. These events primarily correspond to Transact-SQL CREATE, ALTER, and DROP statements, and certain system stored procedures that perform DDL-like operations.

• **Trigger:** CREATE TRIGGER `ap` AFTER INSERT ON `policeofficer` FOR EACH ROW INSERT INTO `pos` VALUES(NEW.OFF_ID,new.STN_ID)

Description: This trigger is triggered when there is an insertion operation in the table policeofficer, the values of officer id and station id are inserted into pos.

• **Trigger:** CREATE TRIGGER `logs` AFTER DELETE ON `criminal`
FOR EACH ROW INSERT INTO `criminallog`(`CRIMINAL_ID`, `AGE`, `GENDER`, `ADDRESS`, `OFFID`, `FIRNO`)VALUES
(old.C_ID,old.C_AGE,old.C_GENDER,old.C_ADDRESS,old.OFF_ID,old.FIR_NO)

Description: This trigger is triggered when there is an deletion operation in the table criminal, the values of deleted row are then inserted into table criminallog.

4.3 PSEUDO CODE

Pseudocode is an informal high-level description of the operating principle of a computer program or other algorithm. It uses the structural conventions of a normal programming language, but is intended for human reading rather than machine reading.

4.3.1 Algorithm for login

Step 1: BEGIN

Step 2: Enter username and password

Step 3: Verify the credentials entered with that in the admin table or policeofficer table

Step 4: If Credentials match, then proceed to the Dashboard page Else show login failed

Step 5: End if

Step 6: END

4.3.2 Algorithm for Table Display

Step 1: BEGIN

Step 2: Establish connection with the database using the username and password of the database.

Step 3: Define section display_table return all the values from the method passed.

Step 4: Define the select query to retrieve all the values from the DBMS

Step 5: Execute query and echo all the values into the table.

Step 6: END

4.3.3 Algorithm for Insert

Step 1: BEGIN

Step 2: Get all the necessary values required for insertion from the variables defined.

Step 3: Define the query for insertion.

Step 4: Execute the Query using the mysqli_query() method.

Step 5: END

4.3.4 Algorithm for update

Step 1: BEGIN

Step 2: Get all the necessary values required for updating from the variables defined.

Step 4: Define the Query for Updating.

Step 5: Execute the Query using the mysqli_query() method defined.

Step 6: END

4.3.5 Algorithm for Delete

Step 1: BEGIN

Step 2: Get the model number of the instrument which is to be deleted into a variable defined in the method.

Step 3: Define the Query for deleting.

Step 4: Execute the Query using the mysqli_query() method.

Step 5: END

4.4 ADMIN.PHP

```php
<?php
session_start();
if(isset($_SESSION["favsport"])){
echo "<script>
window.location.href='logout.php';
alert('Logout of Police?? ');
</script>";
exit();
}
if(isset($_SESSION["favcolor"])){
header("location:admindash.php");
exit();
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>ADMIN</title>
<link rel="stylesheet" href="./styles/style.css">
<link rel="preconnect" href="https://fonts.gstatic.com">
<link             href="https://fonts.googleapis.com/css2?family=Poppins&display=swap"
rel="stylesheet">
</head>
<body>
<header>
<nav>
<div class="logo">
<h4>CRIME STACK</h4>
</div>
<ul class="nav-links">
<li><a href="index.php" >Home</a></li>
<!-- <li><a href="user.php" >User</a></li> -->
<li><a href="police.php" >Police</a></li>
<li class="current"><a href="admin.php">Admin</a></li>
</ul>
</nav>
</header>
<section class="showcase-admin">
<form action="admin.inc.php" method="post">
<div class="login-box">
<h1>Login</h1>
<div class="textbox">
<i class="fas fa-user"></i>
<input type="email" placeholder="Email" name="email" value="">
</div>
<div class="textbox">
```

```
<i class="fas fa-lock"></i>
<input type="password" placeholder="Password" name="passwd" value="">
</div>
<input type="submit" class="btn" name="submitted" value="Sign In">
</div>
</form>
</section>
<footer>
<p>Criminal Record Management System , Copyright &copy 2023</p>
</footer>
</body>
</html>
```

4.5 ADMINDASH.PHP

```php
<?php
session_start();
if(!isset($_SESSION["favcolor"])){
echo "<script>
window.location.href='admin.php';
alert('YOU NEED TO LOGIN FIRST TO ACCESS THIS PAGE!!! ');
</script>";
exit();
}
include_once "dbh.php";
// $sqlc="SELECT * FROM `policeofficer`";
$sqlc="CALL `nfp`();";
$resultc = mysqli_query($conn,$sqlc);
$row1 = mysqli_num_rows($resultc);

$conn->next_result();

// $sqls="SELECT * FROM `station`";
$sqls="CALL `nfs`();";
$result1 = mysqli_query($conn,$sqls);
$row2 = mysqli_num_rows($result1);

$conn->next_result();

// $sqlcr="SELECT * FROM `criminal`";
$sqlcr="CALL `nfc`();";
$resultcr = mysqli_query($conn,$sqlcr);
$row3 = mysqli_num_rows($resultcr);

$conn->next_result();

// $sqlcrime="SELECT * FROM `fir`";
$sqlcrime="CALL `nfcc`();";
$resultcrime = mysqli_query($conn,$sqlcrime);
$row4 = mysqli_num_rows($resultcrime);

$conn->next_result();
?>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Admin | Dashboard</title>
<link rel="stylesheet" href="./styles/style.css">
<link rel="preconnect" href="https://fonts.gstatic.com">
<link           href="https://fonts.googleapis.com/css2?family=Poppins&display=swap"
rel="stylesheet">
```

```
<link          rel="stylesheet"          href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.1/css/all.min.css"                              integrity="sha512-
+4zCK9k+qNFUR5X+cKL9EIR+ZOhtIloNl9GIKS57V1MyNsYpYcUrUeQc9vNfzsWf
V28IaLL3i96P9sdNyeRssA==" crossorigin="anonymous" />
</head>
<body>
<nav>
<div class="logo">
<h4>CRIME STACK</h4>
</div>
<ul class="nav-links">
<li><a href="index.php" >Home</a></li>
<!-- <li><a href="user.php" >User</a></li> -->
<li><a href="police.php" >Police</a></li>
<li class="current"><a href="admin.php">Admin</a></li>
</ul>
</nav>

<section class='showcase-user'>
<aside class='side-menu'>
<ul>
<li          class='current'><i          class="fas          fa-home"></i><a
href="admindash.php">Dashboard</a></li>
<li><i class="fas fa-file-contract"></i><a href="#">Police Station</a>
<ul>
<li><a href="aps.php">Add Police Station</a></li>
<li><a href="manageposstn.php">Manage Police Station</a></li>
</ul>
</li>
<li><i class="fas fa-user-tie"></i><a href="#">Police</a>
<ul>
<li><a href="ap.php">Add Police</a></li>
<li><a href="managepoloff.php">Manage Police</a></li>
</ul>

</li>
<li><i      class="fas      fa-clipboard-list"></i><a      href="viewcriminal.php">View
Criminals</a></li>
<li><i class="fas fa-clipboard-list"></i><a href="viewfir.php">View FIR</a></li>
<li><i class="fas fa-sign-out-alt"></i><a href="logout.php">Log Out</a></li>
</ul>
</aside>
<div class='cards'>
<div class="card">
<i class="fas fa-file sc1"></i>
<h3 class ='sc2'>F I R</h3>
<h1 class='sc3'><?php echo $row4 ; ?></h1>

</div>
<div class="card">
<i class="fas fa-dungeon sc1"></i>
```

```
<h3 class='sc2'>CRIMINALS</h3>
<h1 class='sc3'><?php echo $row3 ; ?></h1>

</div>
<div class="card">
<i class="fas fa-user-tie sc1"></i>
<h3 class='sc2'>POLICE OFFICERS</h3>
<h1 class='sc3'><?php echo $row1 ; ?></h1>

</div>
<div class="card">
<i class="fas fa-building sc1"></i>
<h3 class='sc2'>POLICE STATIONS</h3>
<h1 class='sc3'><?php echo $row2 ; ?></h1>

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

<footer>
<p>Criminal Record Management System , Copyright &copy 2023</p>
</footer>
</body>
</html>
```

4.6 POLICE.PHP

```php
<?php
session_start();
if(isset($_SESSION["favcolor"])){
echo "<script>
window.location.href='logout.php';
alert('Logout of Admin?? ');
</script>";
exit();
}
if(isset($_SESSION["favsport"])){
header("location:policedash.php");
exit();
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>POLICE</title>
<link rel="stylesheet" href="./styles/style.css">
<link rel="preconnect" href="https://fonts.gstatic.com">
<link         href="https://fonts.googleapis.com/css2?family=Poppins&display=swap"
rel="stylesheet">
</head>
<body>
<header>
<nav>
<div class="logo">
<h4>CRIME STACK</h4>
</div>
<ul class="nav-links">
<li><a href="index.php" >Home</a></li>
<!-- <li><a href="user.php" >User</a></li> -->
<li class="current"><a href="police.php" >Police</a></li>
<li><a href="admin.php">Admin</a></li>
</ul>
</nav>
</header>
<section class="showcase-admin">
<form action="police.inc.php" method="post">
<div class="login-box">
<h1>Login</h1>
<div class="textbox">
<i class="fas fa-user"></i>
<input type="text" placeholder="Username" name="user" value="">
</div>
<div class="textbox">
```

```
<i class="fas fa-lock"></i>
<input type="password" placeholder="Password" name="passwd" value="">
</div>
<input type="submit" class="btn" name="submitted" value="Sign In">
</div>
</form>
</section>
<footer>
<p>Criminal Record Management System , Copyright &copy 2023</p>
</footer>
</body>
</html>
```

4.7 POLICEDASH.PHP

```php
<?php
session_start();
if(!isset($_SESSION["favsport"])){
echo "<script>
window.location.href='police.php';
alert('YOU NEED TO LOGIN FIRST TO ACCESS THIS PAGE!!! ');
</script>";
exit();
}
include_once "dbh.php";
// $sqlc="SELECT * FROM `policeofficer`";
$sqlc="CALL `nfp`();";
$resultc = mysqli_query($conn,$sqlc);
$row1 = mysqli_num_rows($resultc);

$conn->next_result();

// $sqls="SELECT * FROM `station`";
$sqls="CALL `nfs`();";
$result1 = mysqli_query($conn,$sqls);
$row2 = mysqli_num_rows($result1);

$conn->next_result();

// $sqlcr="SELECT * FROM `criminal`";
$sqlcr="CALL `nfc`();";
$resultcr = mysqli_query($conn,$sqlcr);
$row3 = mysqli_num_rows($resultcr);

$conn->next_result();

// $sqlcrime="SELECT * FROM `fir`";
$sqlcrime="CALL `nfcc`();";
$resultcrime = mysqli_query($conn,$sqlcrime);
$row4 = mysqli_num_rows($resultcrime);

$conn->next_result();
?>
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Police</title>
<link rel="stylesheet" href="./styles/style.css">
<link rel="preconnect" href="https://fonts.gstatic.com">
<link        href="https://fonts.googleapis.com/css2?family=Poppins&display=swap"
rel="stylesheet">
```

```
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.1/css/all.min.css" integrity="sha512-
+4zCK9k+qNFUR5X+cKL9EIR+ZOhtIloNl9GIKS57V1MyNsYpYcUrUeQc9vNfzsWf
V28IaLL3i96P9sdNyeRssA==" crossorigin="anonymous" />
</head>
<body>
<nav>
<div class="logo">
<h4>CRIME STACK</h4>
</div>
<ul class="nav-links">
<li><a href="index.php" >Home</a></li>
<!--<li><a href="user.php" >User</a></li>-->
<li class="current"><a href="police.php" >Police</a></li>
<li><a href="admin.php">Admin</a></li>
</ul>
</nav>

<section class='showcase-user'>
<aside class='side-menu'>
<ul>
<li class='current'><i class="fas fa-home"></i><a
href="policedash.php">Dashboard</a></li>
<li><i class="fas fa-dungeon"></i><a href="#">Criminals</a>
<ul>
<li><a href="ac.php">Add Criminal</a></li>
<li><a href="managecriminals.php">Manage Criminal</a></li>
</ul>

</li>
<li><i class="fas fa-clipboard-list"></i><a href="#">FIR</a>
<ul>
<li><a href="addfir.php">New FIR</a></li>
<li><a href="managefir.php">View FIR</a></li>
</ul>
</li>
<li><i class="fas fa-clipboard-list"></i><a href="criminallog.php">Criminal
LOG</a></li>
<li><i class="fas fa-clipboard-list"></i><a href="viewcrime.php">Crime
Categories</a></li>
<li><i class="fas fa-sign-out-alt"></i><a href="logout.php">Log Out</a></li>

</ul>
</aside>
<div class='cards'>
<div class="card">
<i class="fas fa-file sc1"></i>
<h3 class ='sc2'>F I R</h3>
<h1 class='sc3'><?php echo $row4 ; ?></h1>

</div>
```

```
<div class="card">
<i class="fas fa-dungeon sc1"></i>
<h3 class='sc2'>CRIMINALS</h3>
<h1 class='sc3'><?php echo $row3 ; ?></h1>

</div>
<div class="card">
<i class="fas fa-user-tie sc1"></i>
<h3 class='sc2'>POLICE OFFICERS</h3>
<h1 class='sc3'><?php echo $row1 ; ?></h1>

</div>
<div class="card">
<i class="fas fa-building sc1"></i>
<h3 class='sc2'>POLICE STATIONS</h3>
<h1 class='sc3'><?php echo $row2 ; ?></h1>

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

<footer>
<p>Criminal Record Management System , Copyright &copy 2023</p>
</footer>
</body>
</html>
<!--<li><i class="fas fa-clipboard-list"></i><a href="#">View Criminals</a></li>-->
```

CHAPTER 5

TESTING

5.1 SOFTWARE TESTING

Testing is the process used to help identify correctness, completeness, security and quality of developed software. This includes executing a program with the intent of finding errors. It is important to distinguish between faults and failures. Software testing can provide objective, independent information about the quality of software and risk of its failure to users or sponsors. It can be conducted as soon as executable software (even if partially complete) exists. Most testing occurs after system requirements have been defined and then implemented in test table programs.

5.2 MODULE TESTING AND INTEGRATION

Module testing is a process of testing the individual subprograms, subroutines, classes, or procedures in a program. Instead of testing whole software program at once, module test in commend testing the small er building blocks of the program. It is largely white box oriented. The objective of doing Module testing is not to demonstrate proper functioning of the module but to demonstrate the presence of an error in the module.

Module testing allow simple of parallel is min to the testing process by giving the opportunity to test multiple modules simultaneously.

The final integrated system too has been tested for various test cases such as duplicate entries and type mismatch.

5.3 APPLICATIONS

- □ To keep record of criminals details for future investigation.
- □ Reduces manual and redundant records keeping.
- □ Keep track of statuses of cases.
- □ Facilitates investigating officer with available of records.
- □ Keeps record of criminals crime activity.

CHAPTER 6

SNAPSHOTS

This chapter consists of working screenshots of the project.

• A home page is the main web page of website. It is a start page of application.
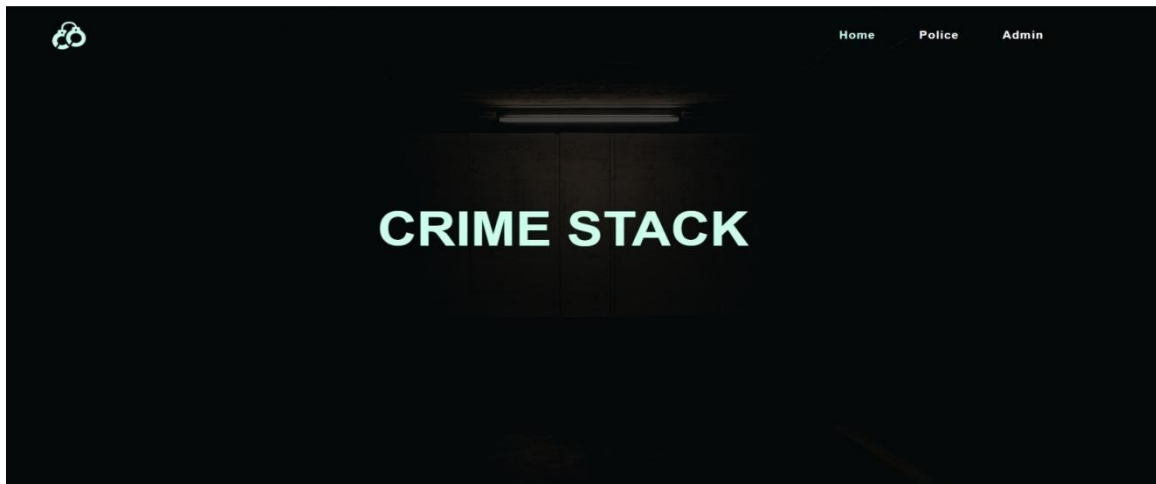
Fig 6.1:Home Page

• A login page is the procedure used to get access to an application. The below page will display when admin, police wants to login.
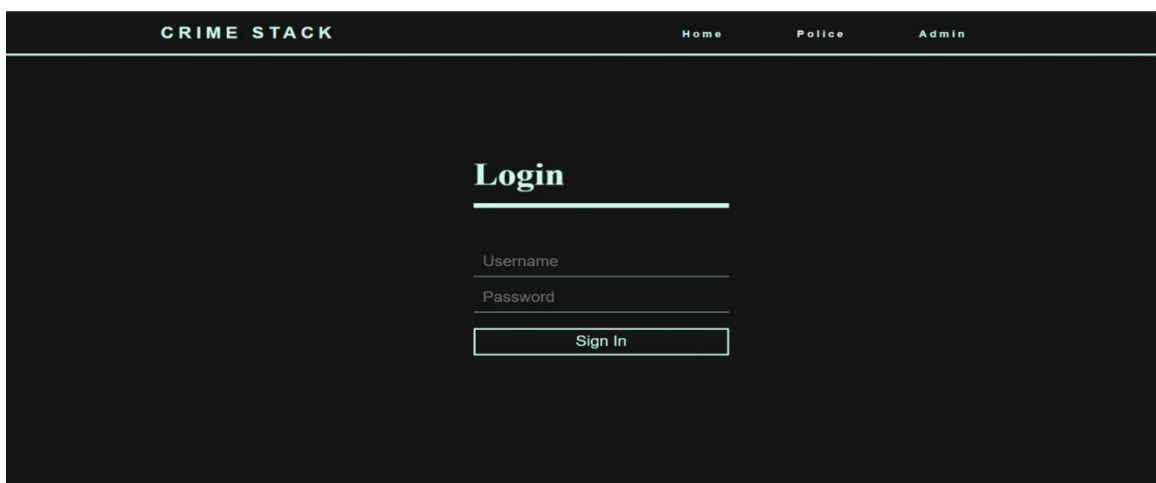
Fig 6.2: Login Page

- Police dashboard contains index of the criminal data reports. It displays when police login.
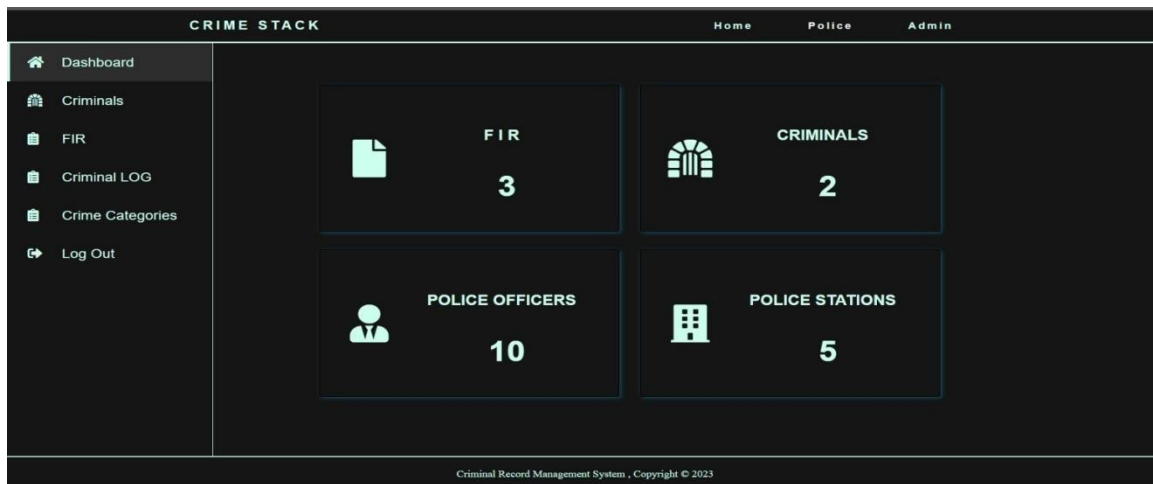
Fig 6.3: Police Dashboard

- Add criminal page helps to add the information about new criminals.

Fig 6.4: Add Criminal Page

- The view criminal section displays the criminal records with officer id.

Fig 6.5: View Criminal Section

● The below page helps to update or modify the criminal details.

Fig 6.6:Update Criminal Details

Fig 6.7:Criminal Log Page

Fig 6.8:Generated Charge Sheet

● A new FIR page is used create a new fir of criminal along with details

Fig 6.9:New FIR Page

● A FIR page helps to view the filed or updated fir report of criminals.

Fig 6.10:View FIR Page

● A update FIR page is used to modify or change the details of already created fir's.

Fig 6.11:Update FIR Page

Fig 6.12:Generated FIR Report

- The below page consists of different types of crimes used to update for criminal

Fig 6.13:List of Crime Categories

- A admin dashboard consists of shortcuts to quick access of information.

Fig6.14:Admin Dashboard

- This page helps to add new police station details.

Fig 6.15:Add Police Station

- The below page consists of list of police stations.

Fig 6.16:List of Police Stations

- The below page is used to update the police station details.

Fig 6.17:Update Police Station

- In this page we can add new police officer information.

Fig 6.18:Add Police Officer

- This page contains the list of all police officers along with the details of police officer.

Fig 6.19:List of Police Officers

- The update page helps to modify or change the information about the police officer.

Fig 6.20:Update Police Officer

CHAPTER 7

CONCLUSION

The **CRIMINAL DATA MANAGEMENT SYSTEM** developed here is the significant of improvement over the manual system which requires a lot of manual works. It also reduces the paperwork to a great extent. This system not only helps countless officers but also increases the speed of all Processes involved, this is very important as it helps to solve cases quickly and provide justice. It can help in digitizing all the information so that it isn't lost in the files. Backlog of cases will be reduced significantly. The system developed here was thoroughly checked and tested by the team and it has proved to be very efficient.

CHAPTER 8

FUTURE ENHANCEMENTS

Future upgrades to this project will implement:

➤ Better interfaces for the ability to view the crime history of the various criminals including FIR report, crimes and information.

➤ However best suited way to deal with the crime specific to particular area system can also be developed on the basis of crime rate analysis.

➤ Ability to change the records of criminals, FIR, police officers and police stations.

➤ Better report implementations between the police officers of different stations.

➤ Ability to see the various crimes of criminal in different police stations.

➤ Ability to view timely criminal reports across various years and months between various time ranges as required

REFERENCES

[1] Ramakrishnan, R., &Gehrke, J. (2011). Database management systems. Boston: McGraw-Hill.

[2] D. K. Tayal, et al., "Crime detection and Criminal identification in India using data Mining Techniques," AI & SOCIETY, vol. 30, No. 1, pp. 117–127, 2015.

[3] Databasesystemsconcepts.EstadosUnidos:McGraw-HillCompanies,Inc.

[4] ShijuSathyadevan,Crimeanalysis and prediction,IEEE,25Sept2014,10.1109/CNSC.20 14.6906719 .

[5] DavidF.(2011).JavaScript:TheDefinitiveGuideSixthedition.

[6] https://www.w3schools.com

[7] https://www.canvasjs.com

[8] https://getbootstrap.com/

[9] https://fontawesome.com