CHAPTER 1

INTRODUCTION

Crime records is concerned with registering a user's complaint. The user can register a complaint with the location and description of the crime which is processed by a police officer of the concerned locality.

PHP is the main software used to create the front end along with CSS and Javascript. MySQL is used for the backend.

1.1 PROBLEM STATEMENT

With the help of crime records the user must be able to register a complaint over a crime committed. The user can first register his/her name, address, aadhar no., gender and mobile no. and create a user id and password. Then the user can use these to login and register a complaint. with the offence, location, date of crime and description.

The police officers can log in and look at the pending and completed complaints. Complaints are assigned to him/her by the police station in charge. He/she can update the status of the case and after completion can close the case.

The police station incharge of each location can add or delete the police officers.They also assign the police officers the cases.

The police head can add police stations in each location. He/She can also view the status of all the complaints.

1.2 OBJECTIVES OF THE PROJECT

The objective of this project is to demonstrate to the user how this program interacts with the system and can register a user's complaint and process it.

The project makes use of interactive programming to enable the user, the police officer, the station incharge and the police headquarters to function in an efficient manner.

1.3 SCOPE OF THE PROJECT

The application program developed can be used in various fields as follows:

- In the field of criminology.
- To aid the police department to manage the records in a more efficient manner.
- To develop a location based police management system.

CHAPTER 2

SYSTEM REQUIREMENTS

A software requirement definition is an abstract description of the services which the system should provide, and the constraints under which the system must operate. It should only specify the external behavior of the system. The requirements are specified as below:

2.1HARDWARE REQUIREMENTS

- **Operating system:** Windows 7 or later
- **Processor:** Intel Pentium 4 or later
- **Memory:** 2GB Minimum ,4GB Recommended
- **Screen Resolution:**1280*1024 or larger
- **Application Window Size:**1024*680 or larger
- **Internet Connection:** Required

2.2 SOFTWARE REQUIREMENTS:

- **Client:** Windows OS
- **Web Server:** XAMPP(Apache Tomcat)
- **Database Server:** MYSQL Server
- **Language:** HTML, PHP , JAVSCRIPT ,CSS

CHAPTER 3

PROBLEM DESCRIPTION

CRIME RECORDS

The police department is managed by the headquarters which takes care of the various police stations in different locations. Each police station has an incharge who manages the various policemen. The user can log into the portal and register a complaint describing the crime. Each policeman can process the case assigned to him by the incharge. The system uses the following tables for maintaining these details:

- COMPLAINT
- HEAD
- POLICE
- POLICE_STATION
- UPDATE_CASE
- USER

The table details are as follows

Table 3.1 - COMPLAINT

COLUMN NAME	DTATYPE & SIZE	CONSTRAINTS	DESCRIPTION
C_ID	INT(11)	PRIMARY KEY	The complaint ID, auto-increments from 1
A_NO	BIGINT(12)	REFERENCES A_NO FROM USER	Aadhar no. of the user, accepts 12 numbers
LOCATION	VARCHAR(50)	NOT NULL	Location of the crime
TYPE_CRIME	VARCHAR(50)	NOT NULL	The type of crime committed
D_O_C	DATE	NOT NULL	Date of occurrence of crime
DESCRIPTION	VARCHAR(50)	NOT NULL	Description of the crime
INC_STATUS	VARCHAR(50)		Status of assignment of the case, i.e., whether it is already assigned to a police officer or not
POL_STATUS	VARCHAR(50)	NULL	Status of the case, i.e., whether the case has been closed and chargesheet has been filed
P_ID	VARCHAR(50)	REFERENCES P_ID FROM POLICE	p_id of the officer assigned to the complaint

Table 3.2 - HEAD

COLUMN NAME	DTATYPE & SIZE	CONSTRAINTS	DESCRIPTION
H_ID	VARCHAR(50)	NOT NULL	The head ID of the head at the police headquarters for logging in
H_PASS	VARCHAR(50)	NOT NULL	The password of the particular head

Table 3.3 - POLICE

COLUMN NAME	DTATYPE & SIZE	CONSTRAINTS	DESCRIPTION
P_NAME	VARCHAR(50)	NOT NULL	Police officer's name
P_ID	VARCHAR(50)	PRIMARY KEY	Police officer's ID for logging in
SPEC	VARCHAR(50)	NOT NULL	Police officer's specialization
LOCATION	VARCHAR(50)	NOT NULL	Location of the police officer's station
P_PASS	VARCHAR(50)	NOT NULL	Police officer's password

Table 3.3 - POLICE_STATION

COLUMN NAME	DTATYPE & SIZE	CONSTRAINTS	DESCRIPTION
I_ID	VARCHAR(50)	PRIMARY KEY	Police station incharge's login ID
I_NAME	VARCHAR(50)	NOT NULL	Station incharge's name
LOCATION	VARCHAR(50)	UNIQUE KEY	Police station's location
I_PASS	VARCHAR(50)	NOT NULL	Station incharge's password

Table 3.3 - CASE_UPDATE

COLUMN NAME	DTATYPE & SIZE	CONSTRAINTS	DESCRIPTION
C_ID	INT(11)	PRIMARY KEY REFERENCES C_ID FROM COMPLAINT	Complaint ID
D_O_U	TIMESTAMP	UNIQUE KEY	Date and time of update of case status
CASE_UPDATE	VARCHAR(200)	NOT NULL	Case status

Table 3.1 - USER

COLUMN NAME	DTATYPE & SIZE	CONSTRAINTS	DESCRIPTION
U_NAME	VARCHAR(50)	NOT NULL	The user's name
U_ID	VARCHAR(50)	UNIQUE KEY	The user's login ID
U_PASS	VARCHAR(50)	NOT NULL	The user's password
U_ADDR	VARCHAR(100)	NOT NULL	The user's address
A_NO	BIGINT(12)	PRIMARY KEY	The user's Aadhar number
GEN	VARCHAR(15)	NOT NULL	The user's gender
MOB	BIGINT(10)	UNIQUE KEY	The user's mobile number

CHAPTER 4

SYSTEM DESIGN

4.1 ER Diagram

Definition:

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how "entities" such as people, objects or concepts relate to each other within a system.

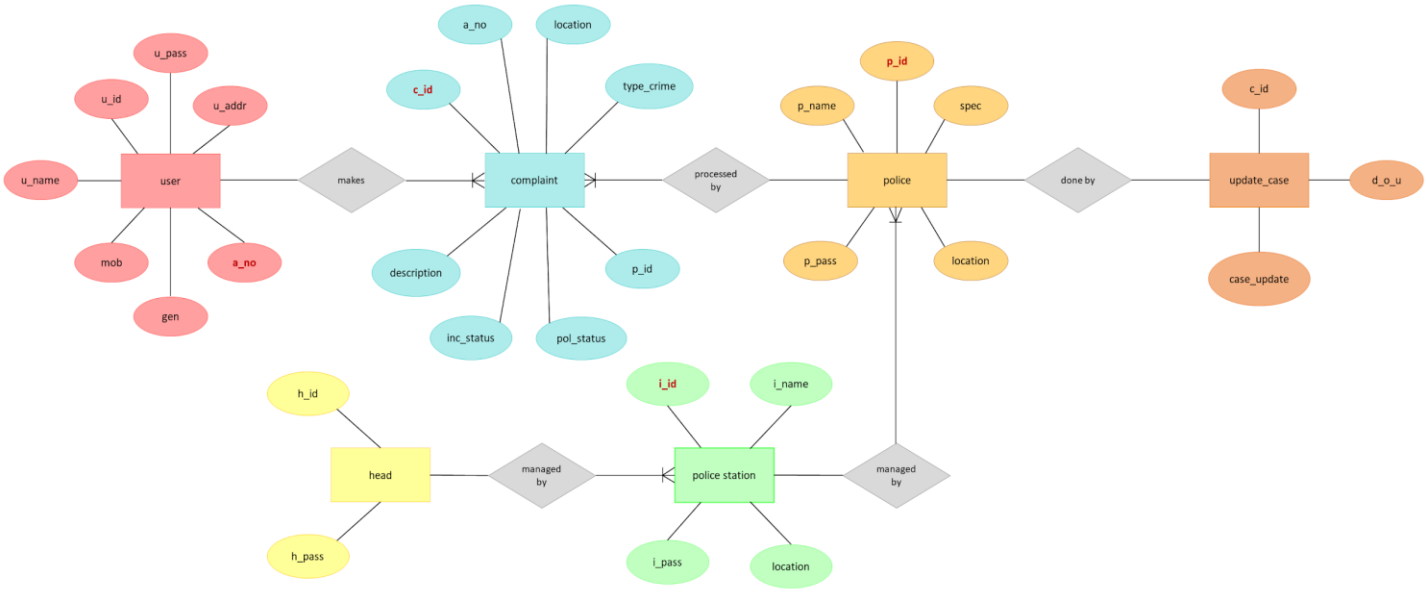

Figure 4.1- ER Diagram for Crime records

4.2 Schema Diagram:

Definition

A schema diagram is a diagram which contains entities and the attributes that will define that schema. A schema diagram only shows us the database design. It does not show the actual data of the database.

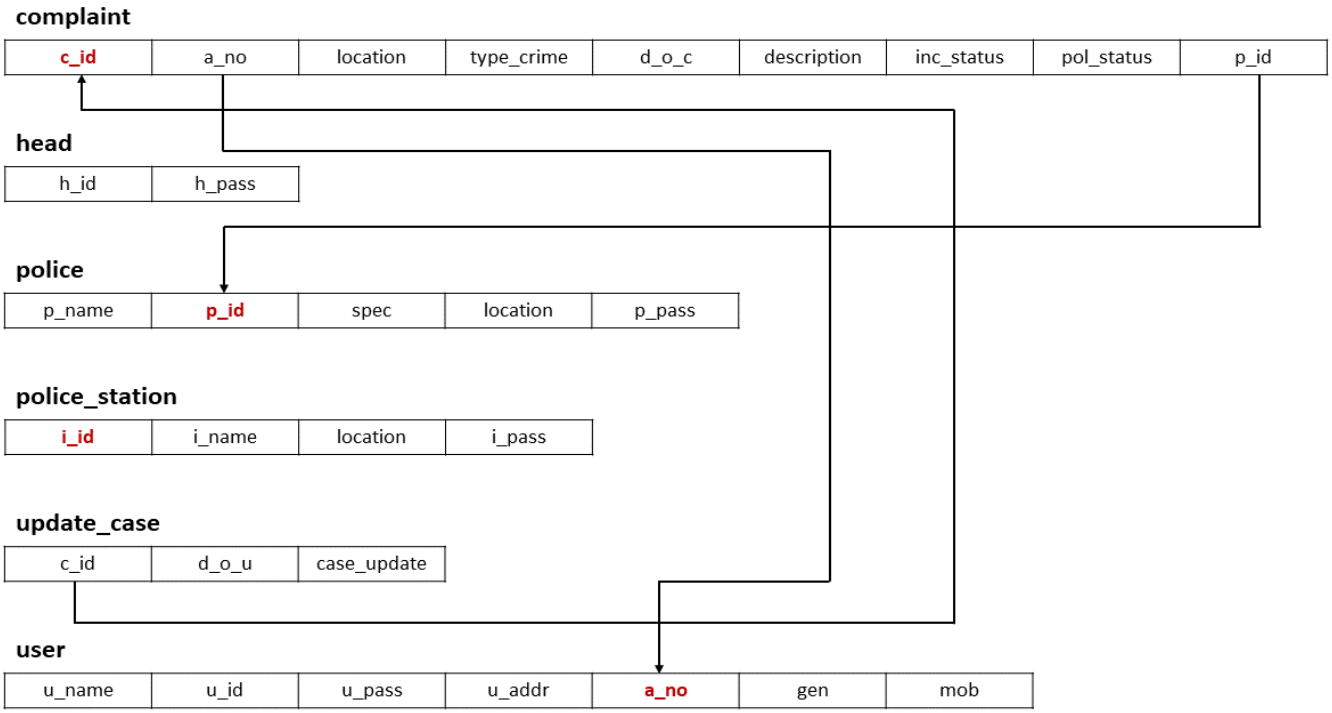

Figure 4.2-Schema diagram for Crime records

4.3 Normal Form

4.3.1 First Normal Form(1NF)

Definition

A relation is in first normal form if and only if the domain of each attribute contains only a single value from that domain.

4.3.2 Second Normal Form(2NF)

Definition

A relation is in the second normal form if it fulfills the following two requirements:

1. It is in first normal form.
2. It does not have any non-prime attribute that is functionally dependent on any proper subset of any candidate key of the relation.

4.3.3 Third Normal Form(3NF)

Definition

A relation is said to meet the third normal form if all the attributes are functionally dependent on solely the primary key.

All the tables in Crime records satisfy all the three normal forms.

CHAPTER 5

IMPLEMENTATION

5.1 Introduction to software used

5.1.1 MySQL

MySQL is an object-relational database management system produced and marketed by Oracle Corporation. MySQL stores data logically in the form of tablespaces and physically in the form of data files ("datafiles"). Tablespaces can contain various types of memory segments, such as Data Segments, Index Segments, etc. Segments in turn comprise one or more extents. Extents comprise groups of contiguous data blocks. Data blocks form the basic units of data storage. A DBA can impose maximum quotas on storage per user within each tablespace. Most Oracle database installations come with a default schema called root. After the installation process sets up sample tables, the user logs into the database with the username root and the password NULL.

5.1.2 PHP

Hypertext Processor (or simply PHP) is a server-side scripting language designed for Web Development, and also used as a general-purpose programming language . PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management systems and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a 21 module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications. The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on web servers on almost every operating system and platform, free of charge.

5.1.3 phpMyAdmin

phpMyAdmin is a free and open source administration tool for MySQL and MariaDB. As a portable web application written primarily in PHP, it has become one of the most popular MySQL administration tools, especially for web hosting services. Features provided by the program include:

• Web interface
• MySQL and MariaDB database
• Import data from CSV and SQL
• Export data to various formats: CSV, SQL, XML, PDF. ISQ/IEC 26300 – OpenDocument Text and Spreadsheet, Word, Excel, LaTex and others
• Searching globally in a database or a subset of it
• Transforming stored data into any format using a set of predefined functions, like displaying BLOB-data as image or download-link 22
• Working with different operating systems
• Make complex SQL queries easier

5.2 Description of Integrated Development Environment

Visual Studio Code is a free source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality. Visual Studio Code is a source-code editor that can be used with a variety of programming languages, including Java, JavaScript, Go, Node.js and C++.

5.3 Source Code

userlogin.php

```
<!DOCTYPE html>
<html>
<head>
<?php

if(isset($_POST['s']))
{
session_start();
$_SESSION['x']=1;
$conn=mysqli_connect("localhost","root","","crime_portal");
if(!$conn)
{
die("could not connect".mysqli_error());
}
mysqli_select_db($conn, "crime_portal");


if($_SERVER["REQUEST_METHOD"]=="POST")
{
$name=$_POST['email'];
$pass=$_POST['password'];
$u_id=$_POST['email'];
$_SESSION['u_id']=$u_id;
$result=mysqli_query($conn,"SELECT u_id,u_pass FROM user where u_id='$name' and
u_pass='$pass' ");
```

```
if(!$result || mysqli_num_rows($result)==0)
{
$message = "Id or Password not Matched.";
echo "<script type='text/javascript'>alert('$message');</script>";
}
else
{
header("location:complainer_page.php");

}
}
}
?>
<link rel="stylesheet" type="text/css"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css">
<link rel="stylesheet" type="text/css" href="https://maxcdn.bootstrapcdn.com/font-
awesome/4.4.0/css/font-awesome.min.css">
<link
href="http://fonts.googleapis.com/css?family=Lato:300,400,700,300italic,400italic,700italic"
rel="stylesheet" type="text/css">

<script>
function f1()
{
var sta2=document.getElementById("exampleInputEmail1").value;
var sta3=document.getElementById("exampleInputPassword1").value;
var x2=sta2.indexOf(' ');
var x3=sta3.indexOf(' ');
if(sta2!="" && x2>=0){
document.getElementById("exampleInputEmail1").value="";
document.getElementById("exampleInputEmail1").focus();
alert("Space Not Allowed");
```

```
}
else if(sta3!="" && x3>=0){
document.getElementById("exampleInputPassword1").value="";
document.getElementById("exampleInputPassword1").focus();
alert("Space Not Allowed");
}

}
</script>

<title>Complainant Login</title>
</head>
<body style="background-size: cover;
background-image: url(regi_bg.jpeg);
background-position: center;">
<nav class="navbar navbar-default navbar-fixed-top" style="height: 60px;">
<div class="container">
<div class="navbar-header">

<a class="navbar-brand" href="home.php" style="margin-top: 5%;"><b>Crime Portal</b></a>
</div>
<div id="navbar" class="collapse navbar-collapse">
<ul class="nav navbar-nav">
<li class="active" style="margin-top: 5%;"><a href="userlogin.php">Complainer Login</a></li>
</ul>

</div>
</div>
</nav>
<div  align="center" >
<div class="form" style="margin-top: 15%">
<form method="post">
```

```
<div class="form-group" style="width: 30%">
<label for="exampleInputEmail1"><h1 style="color: #fff;">User Id</h1></label>
<input type="email" class="form-control" id="exampleInputEmail1" aria-
describedby="emailHelp" size="5" placeholder="Enter Email id" required name="email"
onfocusout="f1()">
</div>
<div class="form-group" style="width:30%">
<label for="exampleInputPassword1"><h1 style="color: #fff;">Password</h1></label>
<input type="password" class="form-control" id="exampleInputPassword1"
placeholder="Password" required name="password" onfocusout="f1()">
</div>

<button type="submit" class="btn btn-primary" name="s" onclick="f1()">Submit</button>
</form>
</div>
</div>
<div style="position: fixed;
left: 0;
bottom: 0;
width: 100%;
background-color: rgba(0,0,0,0.7);
color: white;
text-align: center;">
<h4 style="color: white;">&copy <b>Crime Records</b></h4>
</div>

</body>
</html>
```

complainer_page.php

```
<!DOCTYPE html>
<html>

<?php
session_start();
if(!isset($_SESSION['x']))
header("location:userlogin.php");



$conn=mysqli_connect("localhost","root","","crime_portal");
if(!$conn)
{
die("could not connect".mysqli_error());
}
mysqli_select_db($conn, "crime_portal");

$u_id=$_SESSION['u_id'];

$result=mysqli_query($conn,"SELECT a_no FROM user where u_id='$u_id' ");
$q2=mysqli_fetch_assoc($result);
$a_no=$q2['a_no'];

$result1=mysqli_query($conn,"SELECT u_name FROM user where u_id='$u_id' ");
$q2=mysqli_fetch_assoc($result1);
$u_name=$q2['u_name'];



if(isset($_POST['s'])){
$con=mysqli_connect('localhost','root','');
if(!$con)
```

```
{
die('could not connect: '.mysqli_error());
}
if($_SERVER["REQUEST_METHOD"]=="POST")
{



$location=$_POST['location'];
$type_crime=$_POST['type_crime'];
$d_o_c=$_POST['d_o_c'];
$description=$_POST['description'];

$var=strtotime(date("Ymd"))-strtotime($d_o_c);



if($var>=0)
{

$comp="INSERT into complaint(a_no,location,type_crime,d_o_c,description)
values('$a_no','$location','$type_crime','$d_o_c','$description')";
mysqli_select_db($conn,"crime_portal");
$res=mysqli_query($conn,$comp);

if(!$res)
{
$message1 = "Complaint already filed";
echo "<script type='text/javascript'>alert('$message1');</script>";
}
else
{
$message = "Complaint Registered Successfully";
echo "<script type='text/javascript'>alert('$message');</script>";
```

```
}
}
else
{
$message = "Enter Valid Date";
echo "<script type='text/javascript'>alert('$message');</script>";
}
}
}
?>

<script>
function f1()
{
var sta1=document.getElementById("desc").value;
var x1=sta1.trim();
if(sta1!="" && x1==""){
document.getElementById("desc").value="";
document.getElementById("desc").focus();
alert("Space Found");
}
}
</script>

<head>
<title>Complainer Home Page</title>

<link rel="stylesheet" type="text/css"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/css/bootstrap.min.css">

<link rel="stylesheet" type="text/css" href="https://maxcdn.bootstrapcdn.com/font-
awesome/4.4.0/css/font-awesome.min.css">
```

```
<link
href="http://fonts.googleapis.com/css?family=Lato:300,400,700,300italic,400italic,700italic"
rel="stylesheet" type="text/css">

<link href="complainer_page.css" rel="stylesheet" type="text/css" media="all" />

</head>

<body style="background-size: cover;
background-image: url(home_bg1.jpeg);
background-position: center;">
<nav  class="navbar navbar-default navbar-fixed-top">
<div class="container">
<div class="navbar-header">
<button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-
target="#navbar" aria-expanded="false" aria-controls="navbar">
<span class="sr-only">Toggle navigation</span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
<span class="icon-bar"></span>
</button>
<a class="navbar-brand" href="home.php"><b>Home</b></a>
</div>
<div id="navbar" class="collapse navbar-collapse">
<ul class="nav navbar-nav">
<li ><a href="userlogin.php">User Login</a></li>
<li class="active"><a href="complainer_page.php">User Home</a></li>
</ul>

<ul class="nav navbar-nav navbar-right">
<li class="active"><a href="complainer_page.php">Log New Complain</a></li>
```

```
<li><a href="complainer_complain_history.php">Complaint History</a></li>
<li><a href="logout.php">Logout   <i class="fa fa-sign-out" aria-
hidden="true"></i></a></li>
</ul>
</div>
</div>
</nav>


<div class="video" style="margin-top: 5%">
<div class="center-container">
<div class="bg-agile">
<br><br>
<div class="login-form"><p><h2 style="color:white">Welcome <?php echo "$u_name"
?></h2></p><br>
<p><h2>Log New Complain</h2></p><br>
<form action="#" method="post" style="color: gray">Aadhar
<input type="text"  name="aadhar_number" placeholder="Aadhar Number" required="" disabled
value=<?php echo "$a_no"; ?>>

<div class="top-w3-agile" style="color: gray">Location of Crime

<select class="form-control" name="location">
<?php
$loc=mysqli_query($conn,"select location from police_station");
while($row=mysqli_fetch_array($loc))
{
?>
<option> <?php echo $row[0]; ?> </option>
<?php
}
?>
```

```
</select>
</div>
<div class="top-w3-agile" style="color: gray">Type of Crime
<select class="form-control" name="type_crime">
<option>Theft</option>
<option>Robbery</option>
<option>Pick Pocket</option>
<option>Murder</option>
<option>Rape</option>
<option>Molestation</option>
<option>Kidnapping</option>
<option>Missing Person</option>
</select>
</div>
<div class="Top-w3-agile" style="color: gray">
Date Of Crime :    
<input style="background-color: #313131;color: white" type="date" name="d_o_c" required>
</div>
<br>
<div class="top-w3-agile" style="color: gray">
Description
<textarea  name="description" rows="20" cols="50" placeholder="Describe the incident in details
with time" onfocusout="f1()" id="desc" required></textarea>
</div>
<input type="submit" value="Submit" name="s">
</form>
</div>
</div>
</div>
</div>
<div style="position: relative;
```

```
left: 0;
bottom: 0;
width: 100%;
height: 30px;
background-color: rgba(0,0,0,0.8);
color: white;
text-align: center;">
<h4 style="color: white;">&copy <b>Crime Records</b></h4>
</div>
<script type="text/javascript" src="https://code.jquery.com/jquery-2.1.4.js"></script>
<script type="text/javascript"
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.5/js/bootstrap.min.js"></script>
</body>
</html>
```

5.4 SQL Trigger

TRIGGER TO UPDATE COMPLAINT WHEN DELETING POLICE OFFICER

```
DELIMITER $$
CREATE TRIGGER Sqltrig BEFORE DELETE ON police
FOR EACH ROW BEGIN
UPDATE complaint
SET complaint.p_id = 'Null'
AND complaint.inc_status = 'Unassigned'
WHERE complaint.p_id = police.p_id
END;
$$
DELIMITER
```

TRIGGER TO CONVERT USER'S NAME TO UPPER CASE

```
DELIMITER $$
CREATE TRIGGER Sqltrig BEFORE INSERT ON user
FOR EACH ROW SET NEW.u_name=UPPER(NEW.u_name);
$$
DELIMITER
```

CHAPTER 6

SAMPLE SCREEN SHOTS

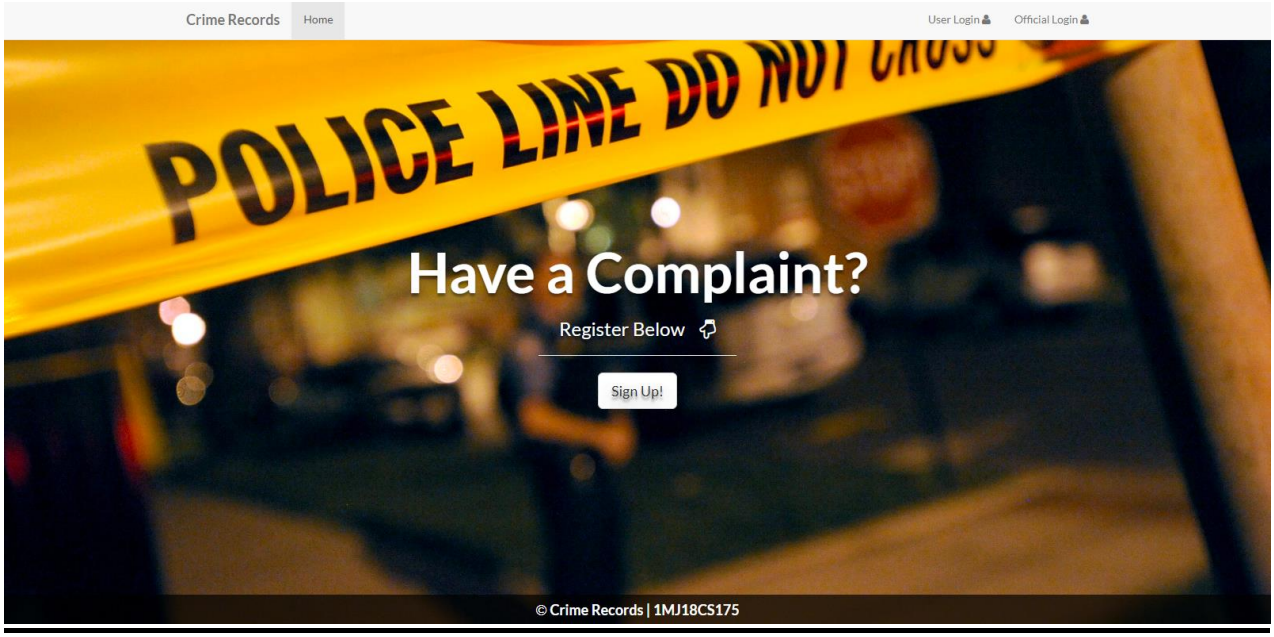

Figure 7.1 HOME PAGE

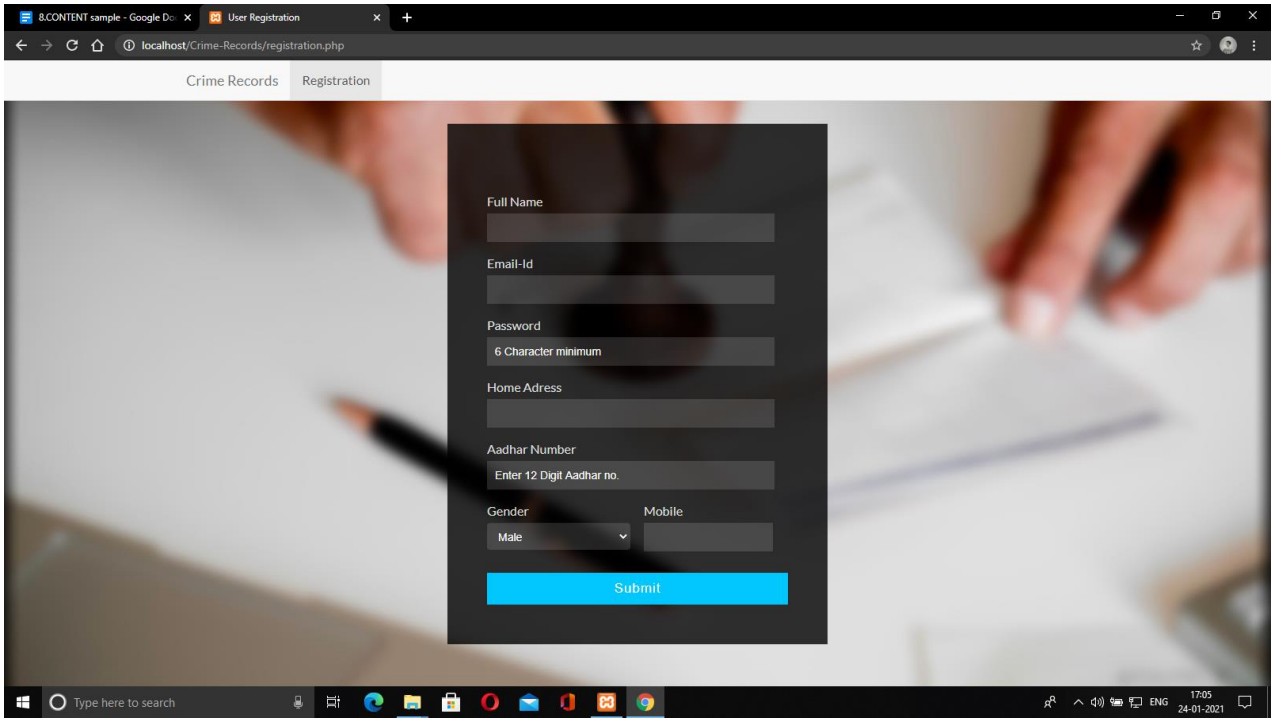

Figure 7.2 USER REGISTRATION

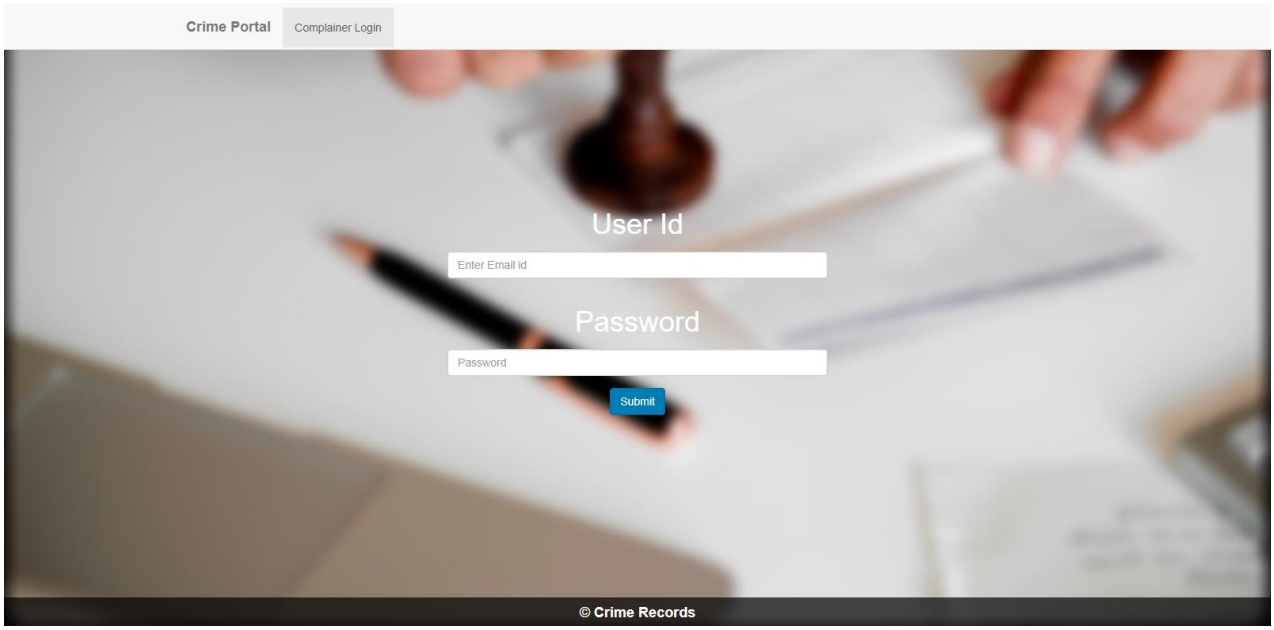

Figure 7.3 USER LOGIN

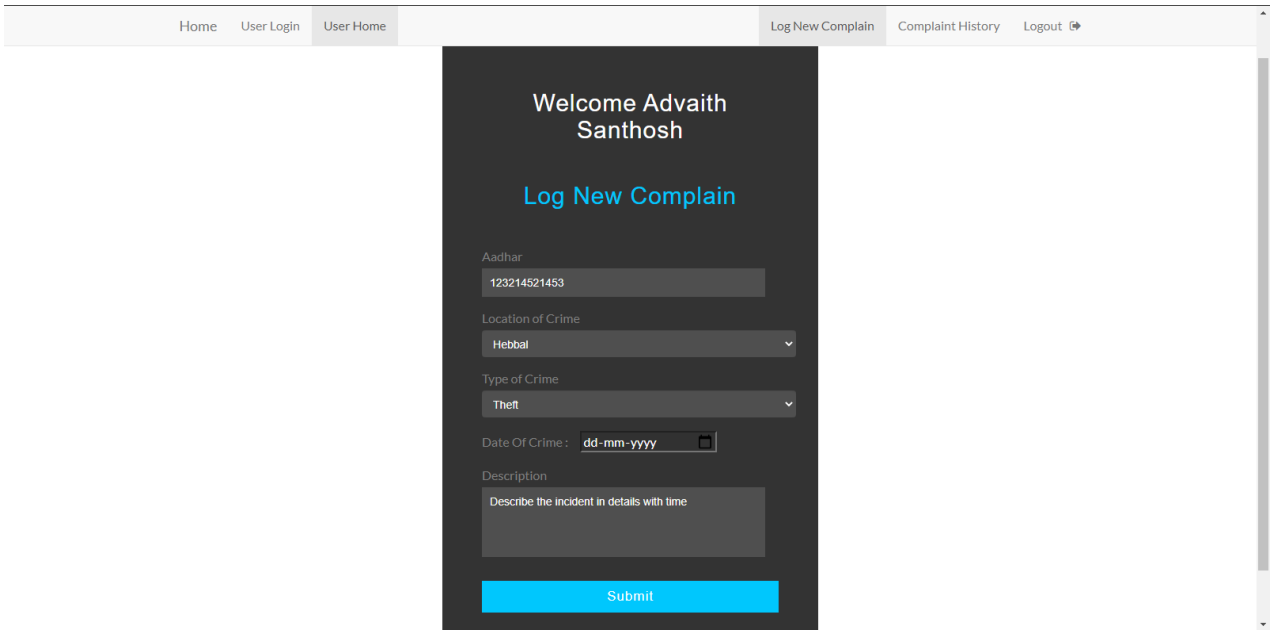

Figure 7.4 COMPLAINT PAGE

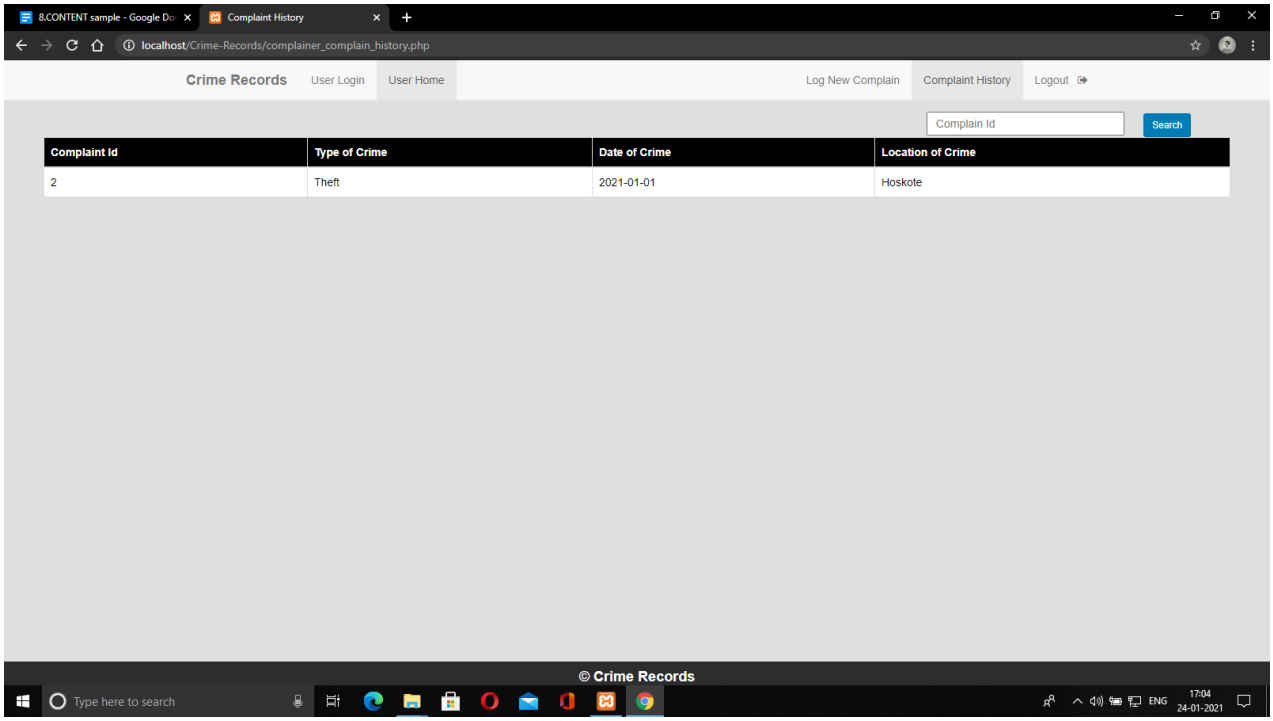

Figure 7.5 USER COMPLAINT HISTORY

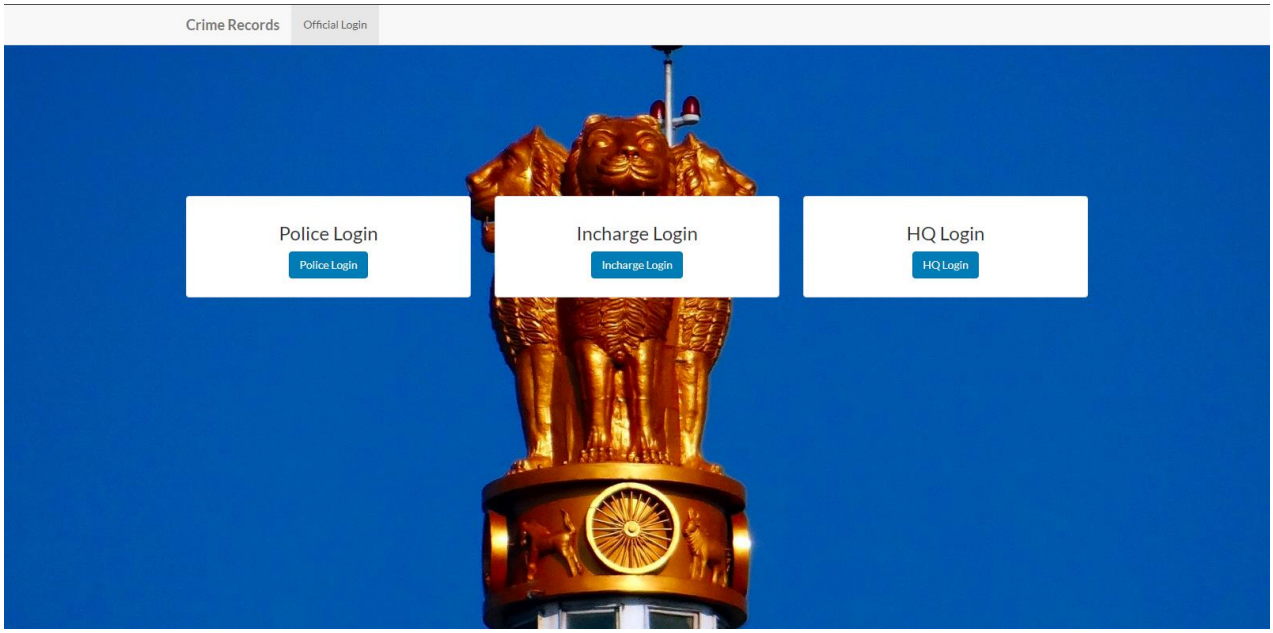

Figure 7.6 OFFICIAL LOGIN

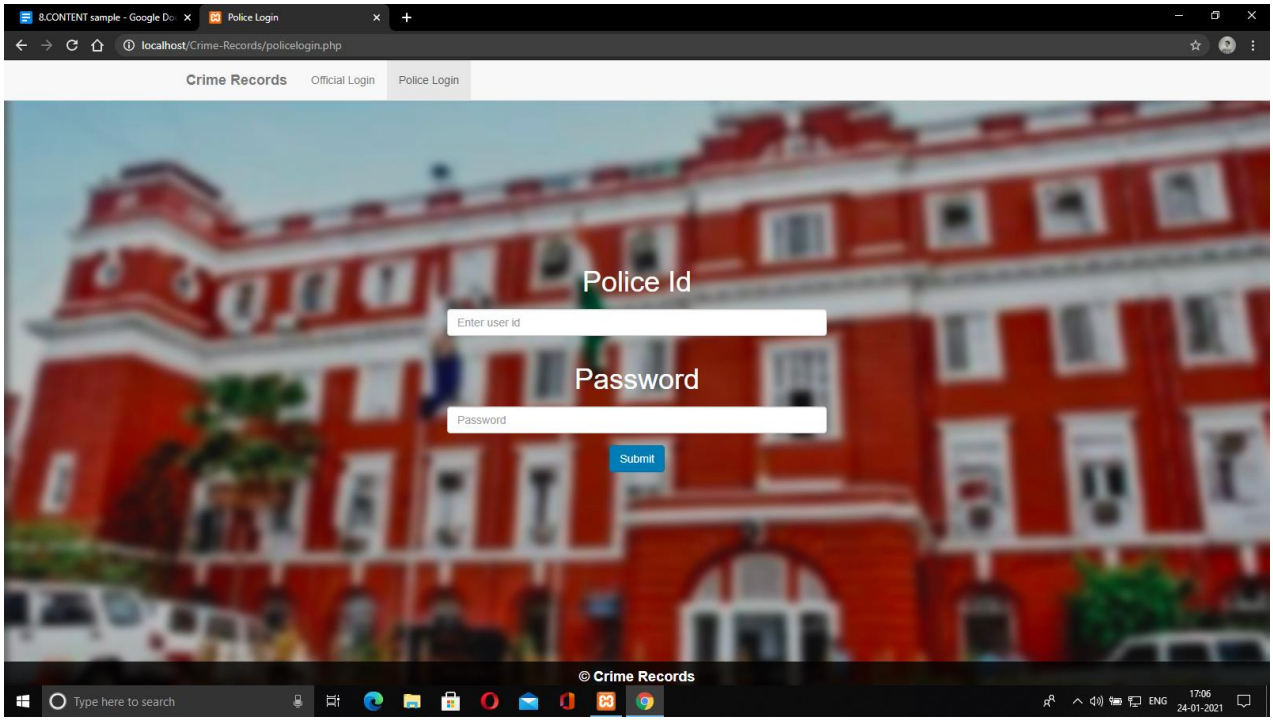

Figure 7.7 POLICE LOGIN

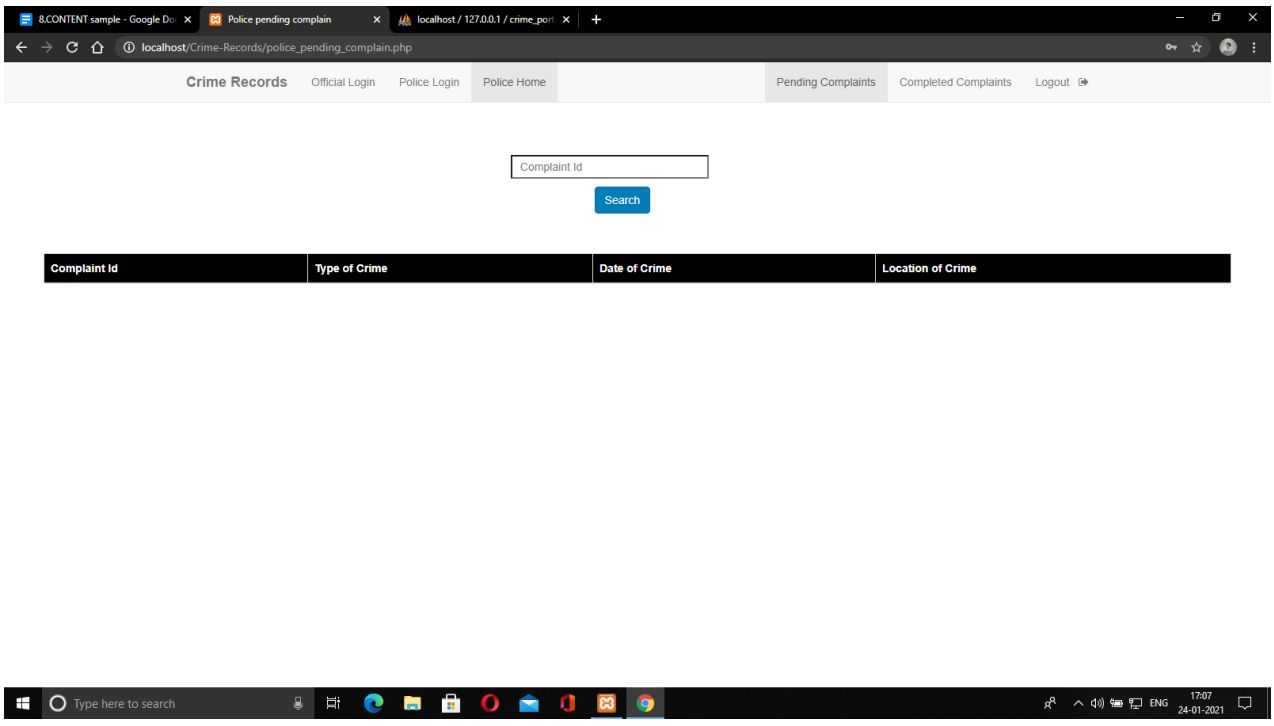

Figure 7.8 POLICE PENDING COMPLAINTS

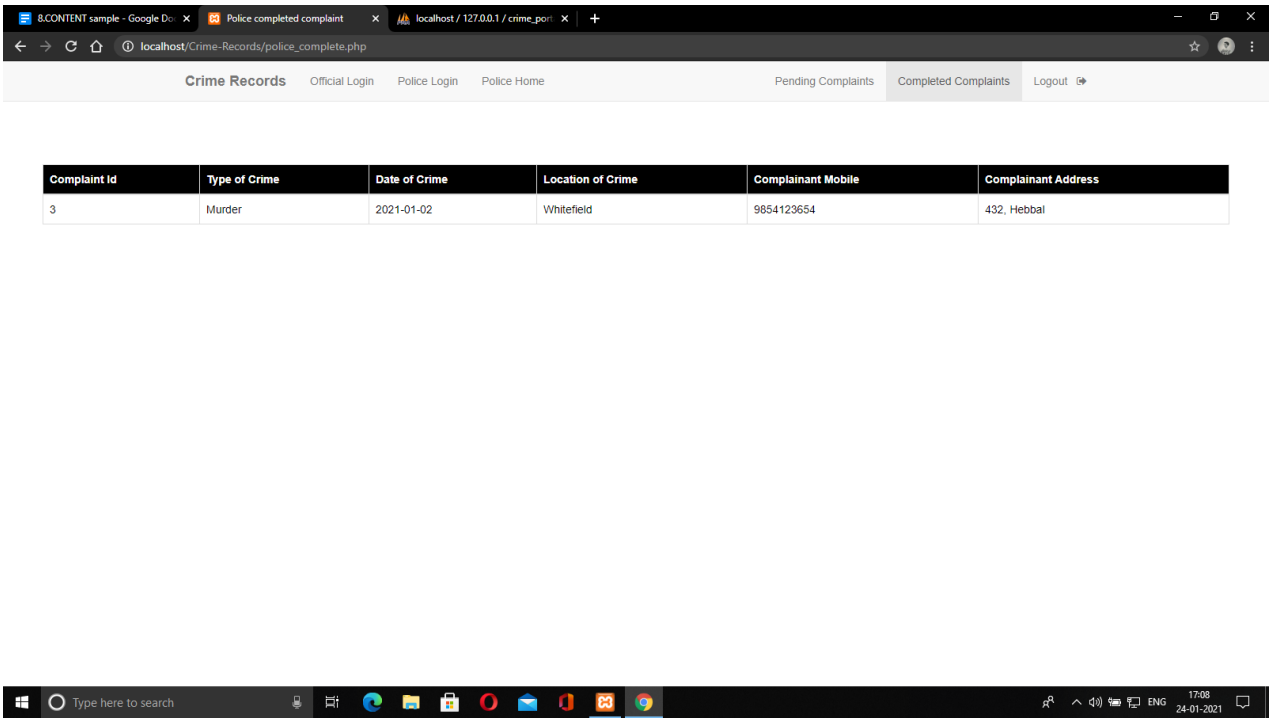

Figure 7.9 POLICE COMPLETED COMPLAINTS

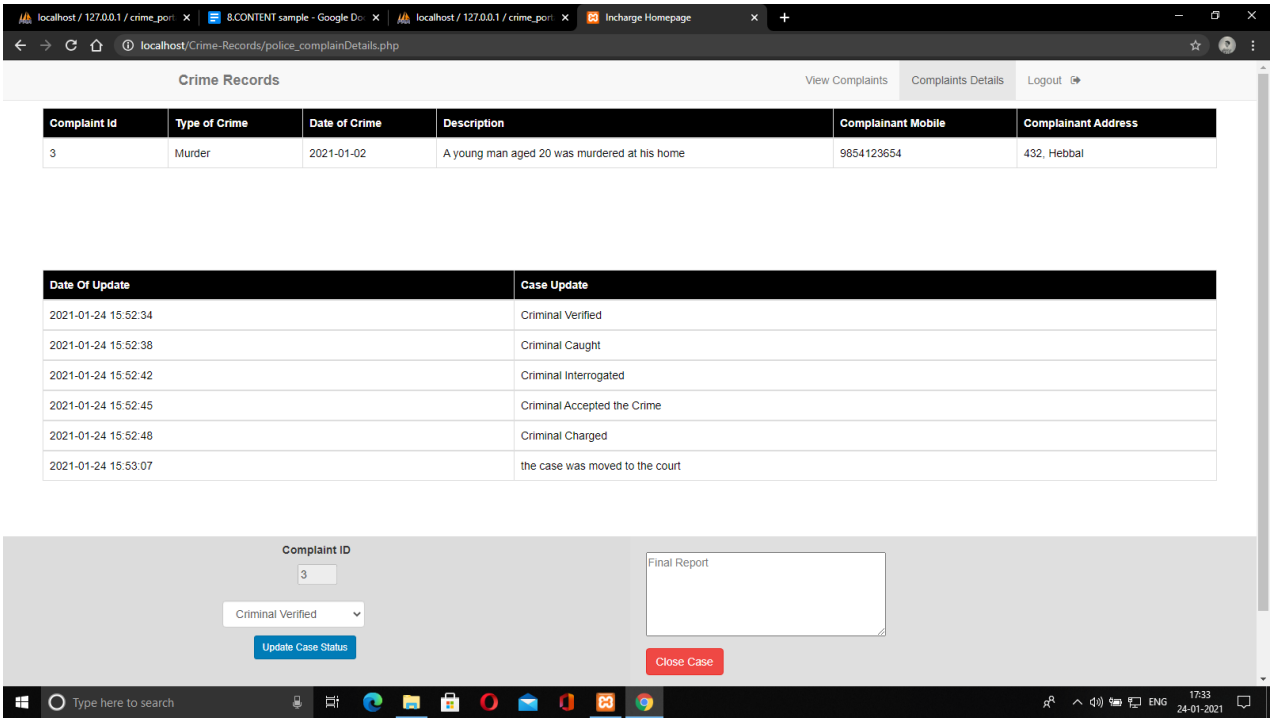

Figure 7.10 POLICE CASE STATUS UPDATE

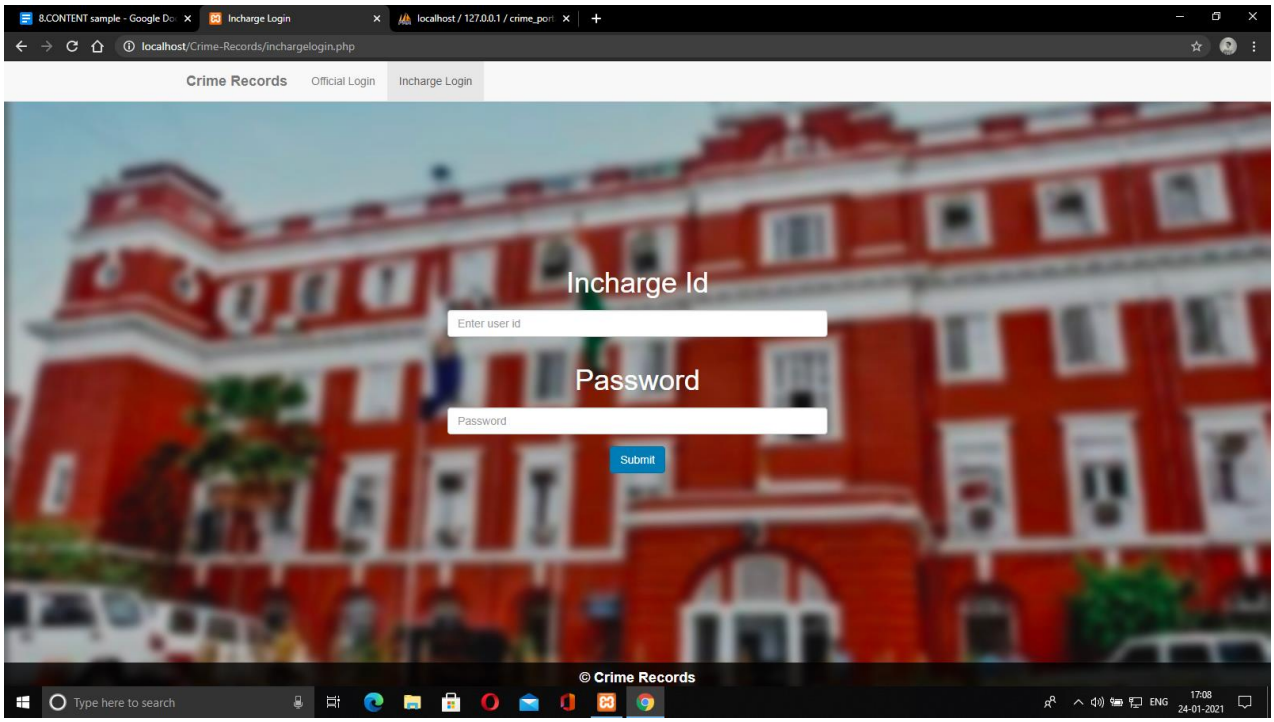

Figure 7.11 STATION INCHARGE LOGIN

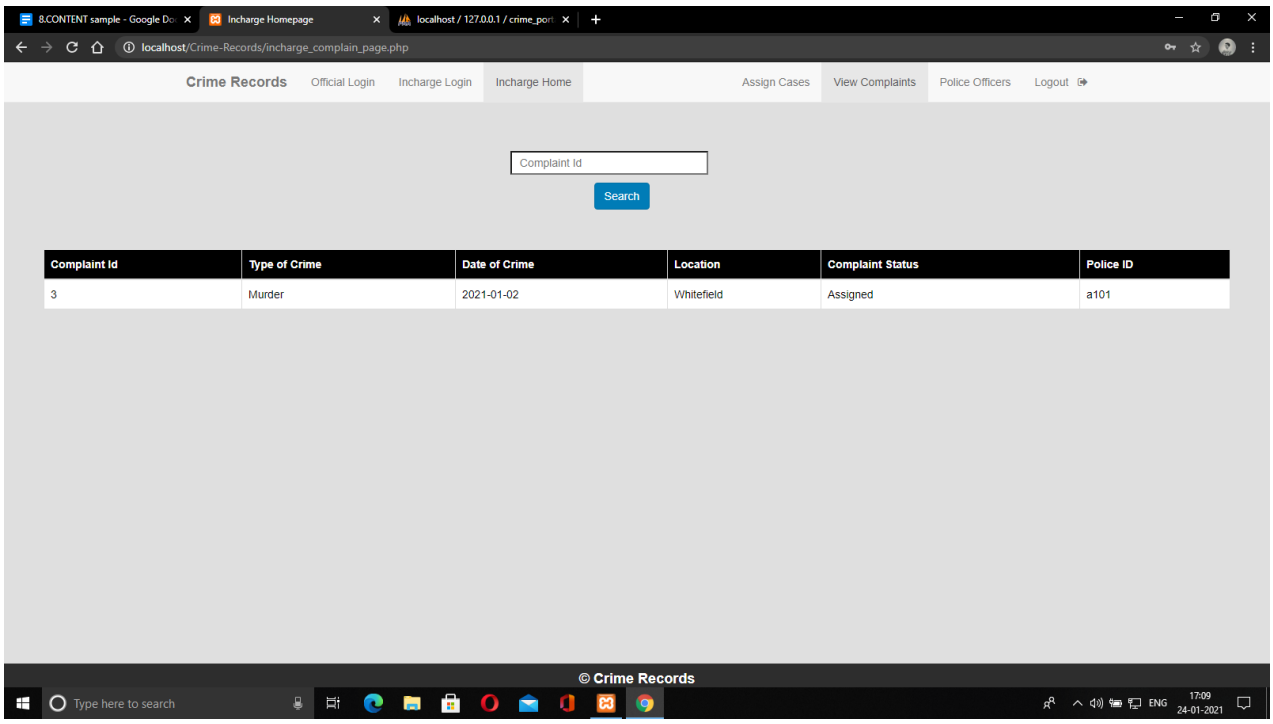

Figure 7.12 INCHARGE VIEW COMPLAINTS

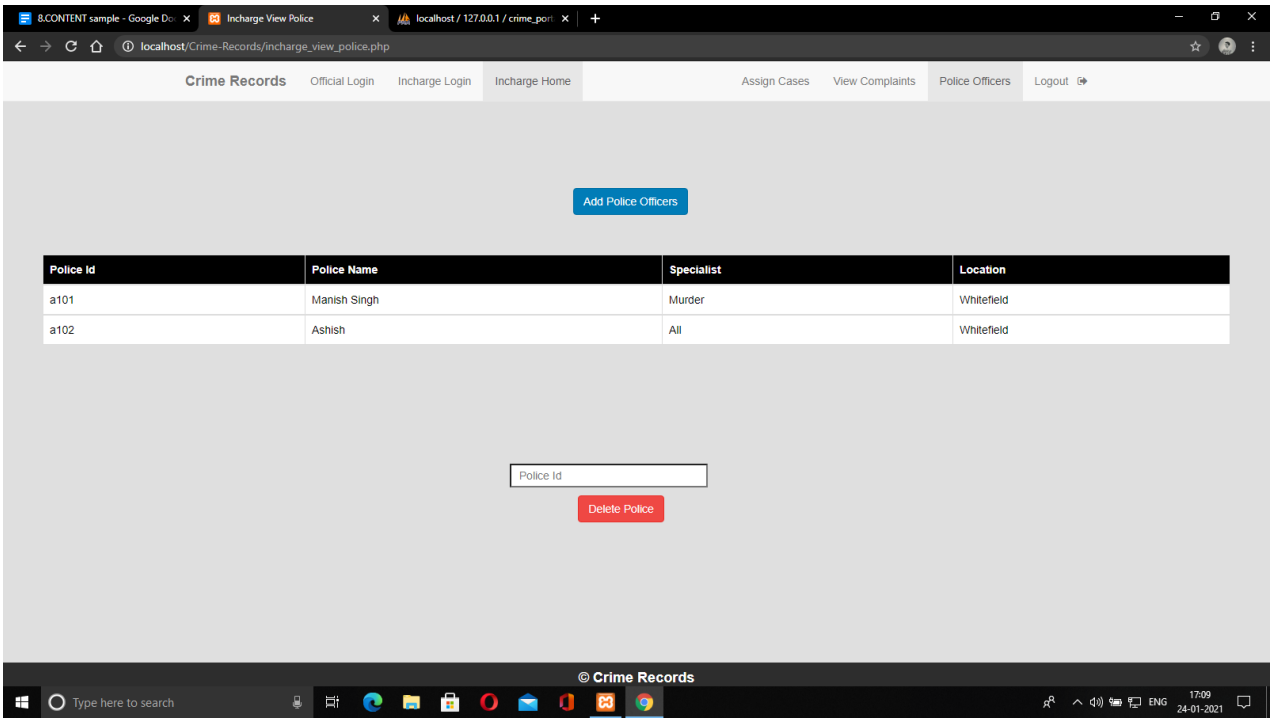

Figure 7.13 INCHARGE VIEW POLICE OFFICERS

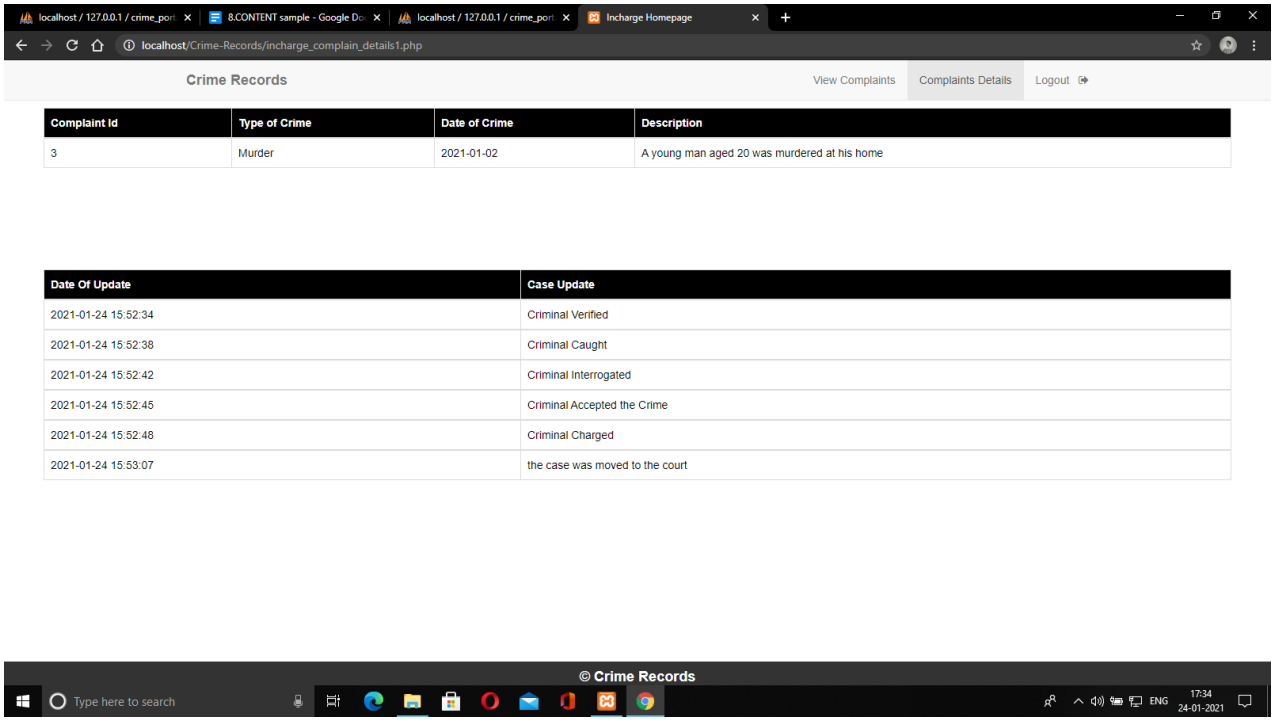

Figure 7.14 INCHARGE COMPLAINT DETAILS

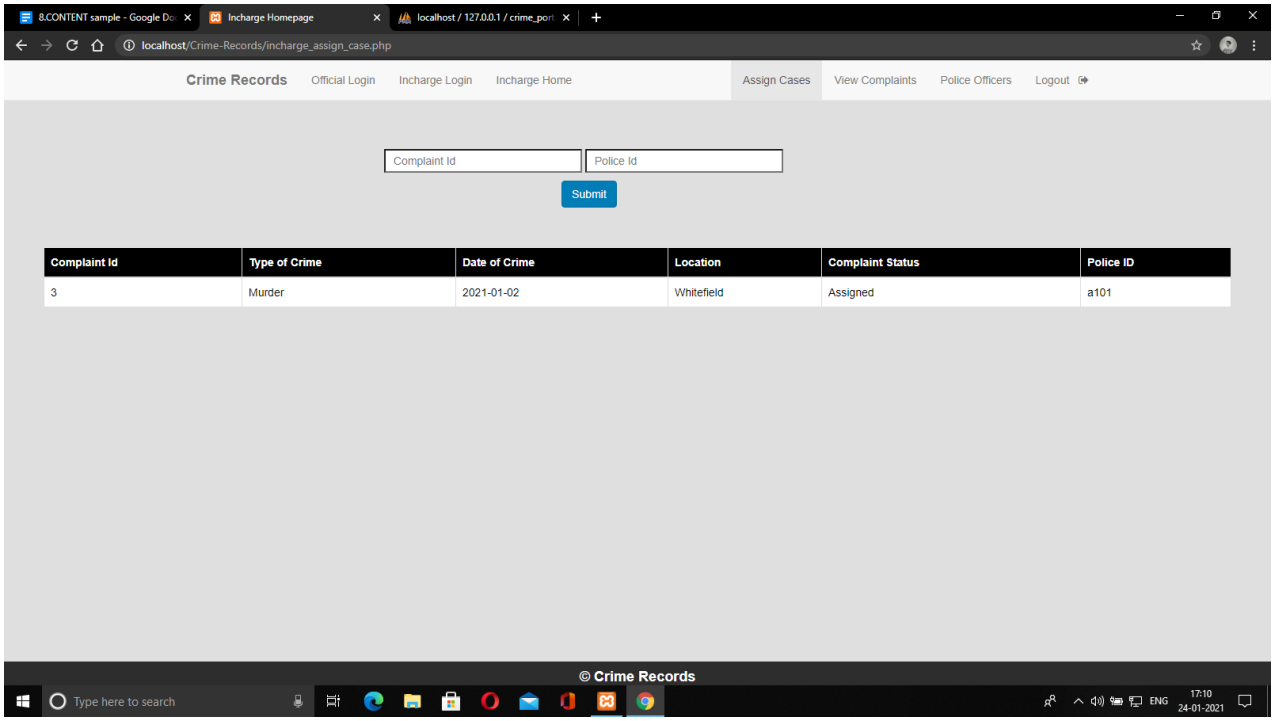

Figure 7.15 INCHARGE ASSIGN CASES

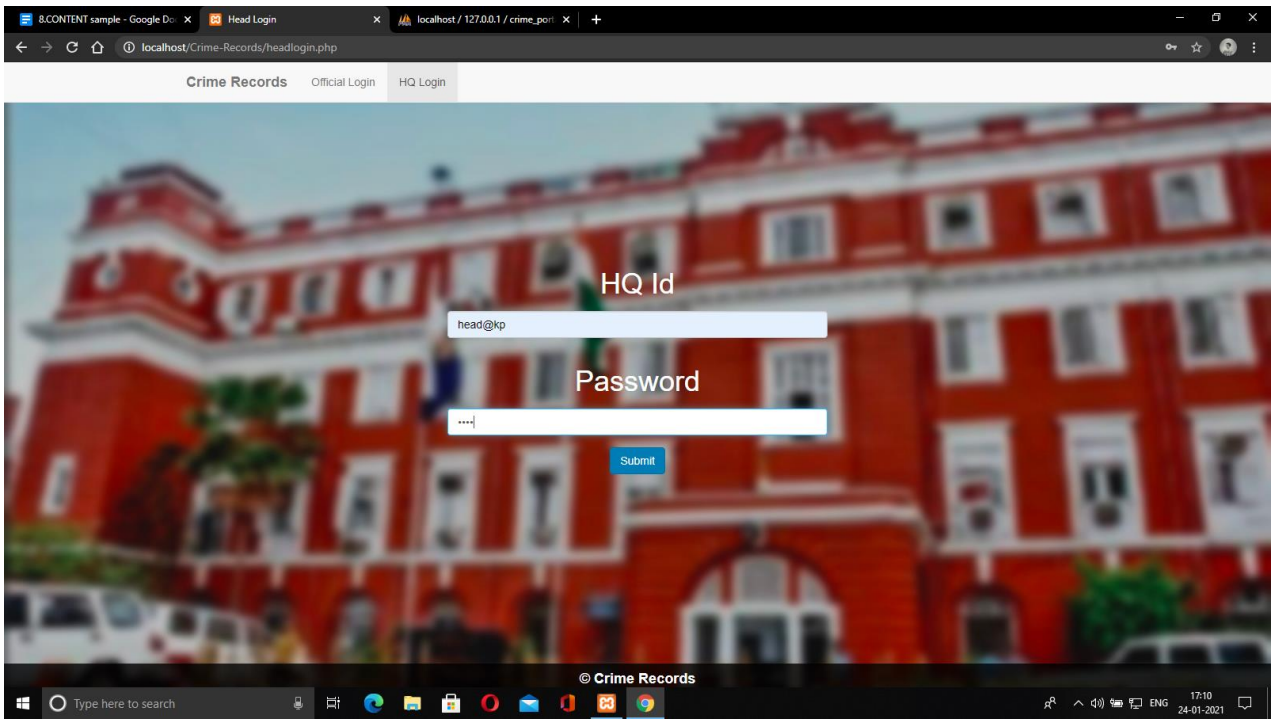

Figure 7.16 POLICE HEAD LOGIN

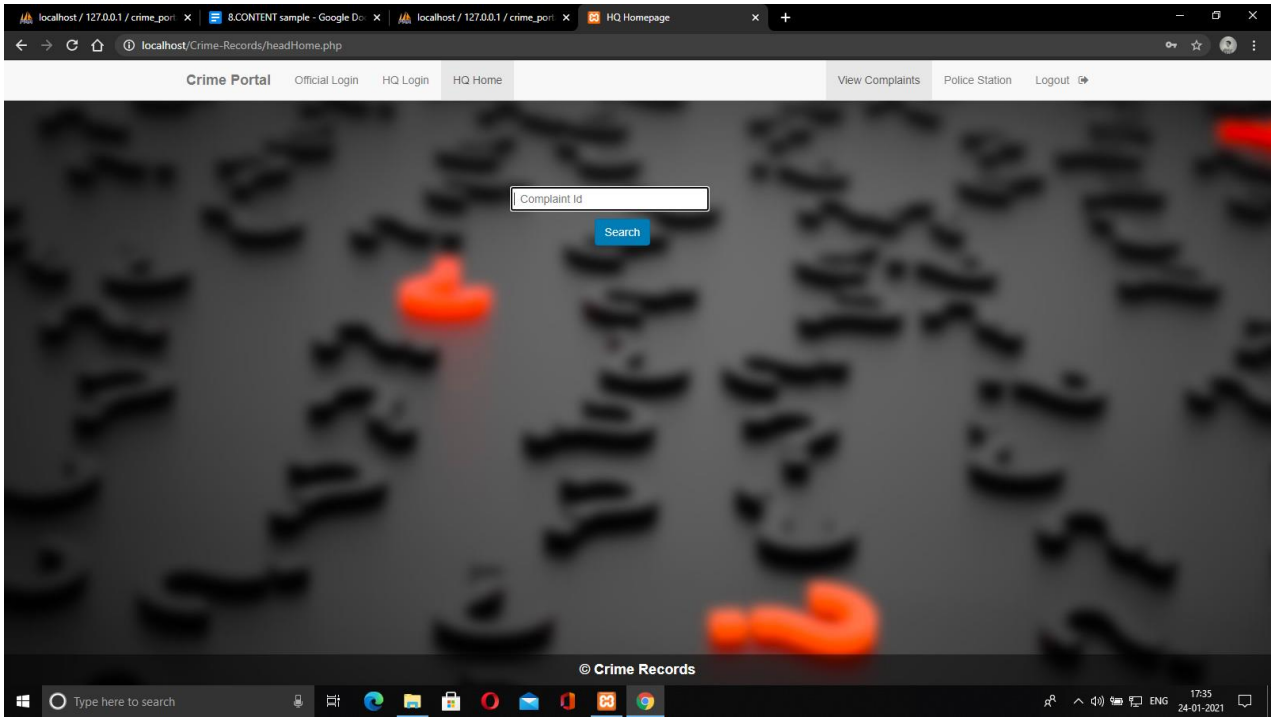

Figure 7.17 POLICE HEAD COMPLAINT SEARCH

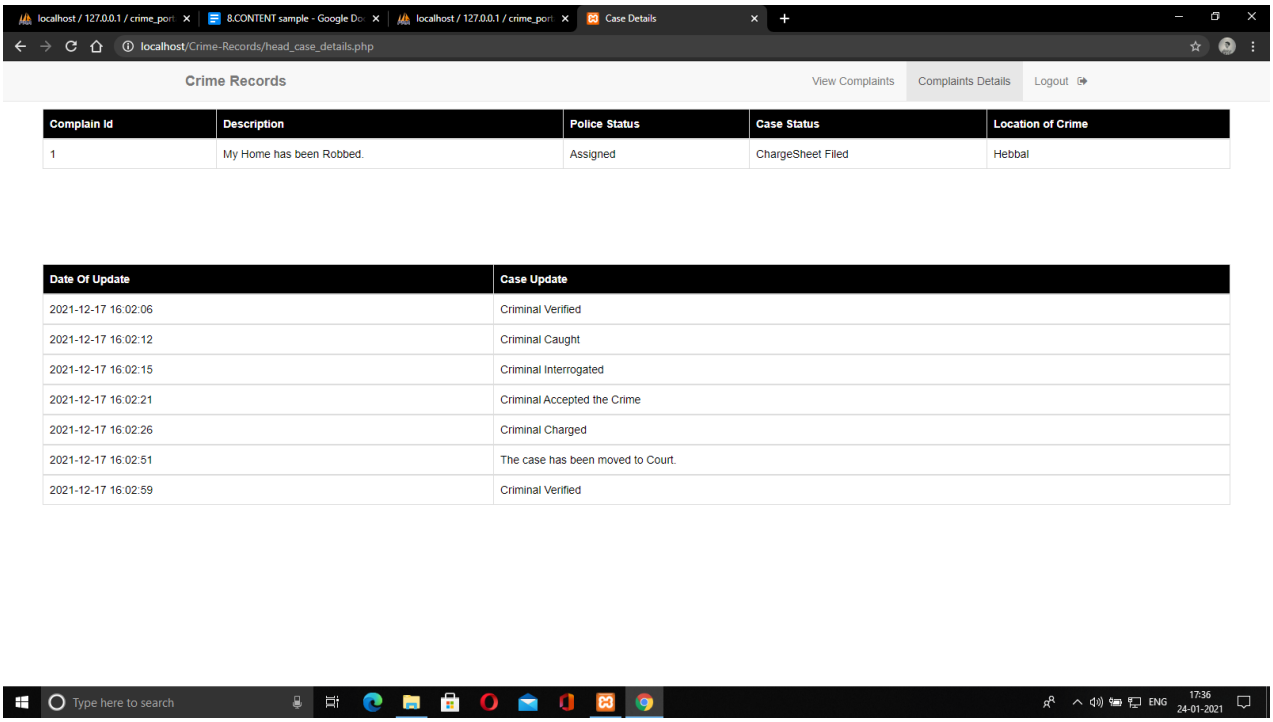

Figure 7.18 POLICE HEAD COMPLAINT DETAILS

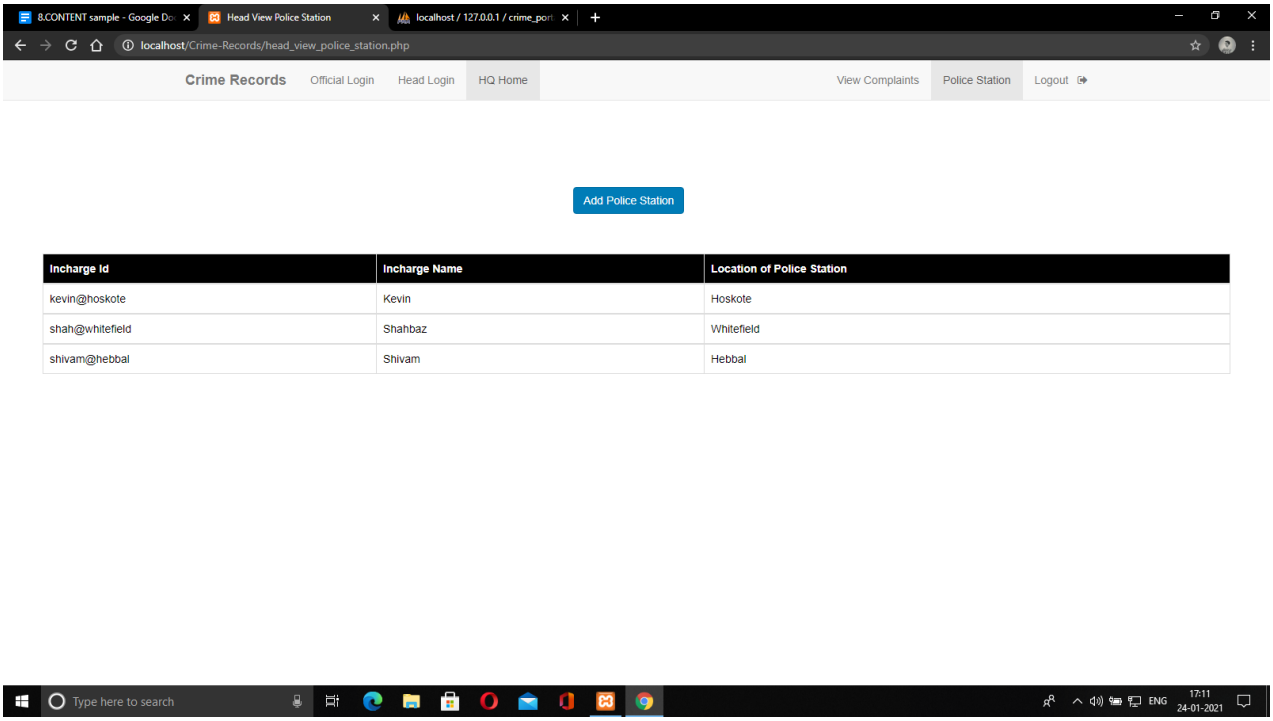

Figure 7.19 POLICE HEAD VIEW POLICE STATIONS

CHAPTER 7

CONCLUSION & FUTURE SCOPE

It has been a great pleasure for me to work on this exciting and challenging project. This project proved good for me as it provided practical knowledge of not only programming in PHP and MySQL web based application.This project has made me aware of the immense capabilities and the various uses of PHP, CSS, MySQL, Apache server both individually and combined. It also provides knowledge about the latest technology used in developing web enabled application and client server technology that will be in great demand in future. This will provide better opportunities and guidance in future in developing projects independently.

.

REFERENCES

1. Explore PHP
   - ➢ Dave Grundgeiger
   - ➢ Publisher: O' Reilly

2. Apache Server Basics
   - ➢ Roger S Pressman
   - ➢ McGraw-Hill Publishing

3. Web Programming
   - ➢ W.B Sabesta
   - ➢ 3 rd Edition

4. http://www.apache.net
5. http://www.w3schools.com
6. http://www.databasemysql.com
7. http://www.php.com