FEDERAL COURT DATABASE

Abstract

In this current period situation, where many crimes are committed it is being difficult for the lawyers and judges to maintain records of different cases. So there is a lot of misconception going on in and around the world. As a result, of incorrect information about different cases many innocent people are being sentenced as criminals even though they don't deserve to be criminals. Henceforth, many people fear to get any kind of judgment. Hence this project can be useful to store information about different cases, criminals details, lawyers details, judge details and which in turn helps to handle different cases efficiently .

Introduction

In this project 'Federal court database' we are trying to maintain an record about each and every case held in a particular court. All the details of each and every case held in that court can be maintained in an safe and secure way. This project also helps us to prevent the loss of records due to any physical damage like robbery.

In this project we will be providing each and every details of the client who's case is to be solved, the lawyer who takes up the case of client, the judge who gives the judgment and all the other details of the case can be viewed.

FEDERAL COURT DATABASE

Hardware/software requirements

HARDWARE- A Basic Desktop or Laptop(Ram 256MB,HDD-40GB)

SOFTWARE- Frontend : Personal Home Page(Php).

Backend : My sql

Server Configuration :

Apache Version :2.4.9 - Documentation

PHP Version :5.5.12 - Documentation

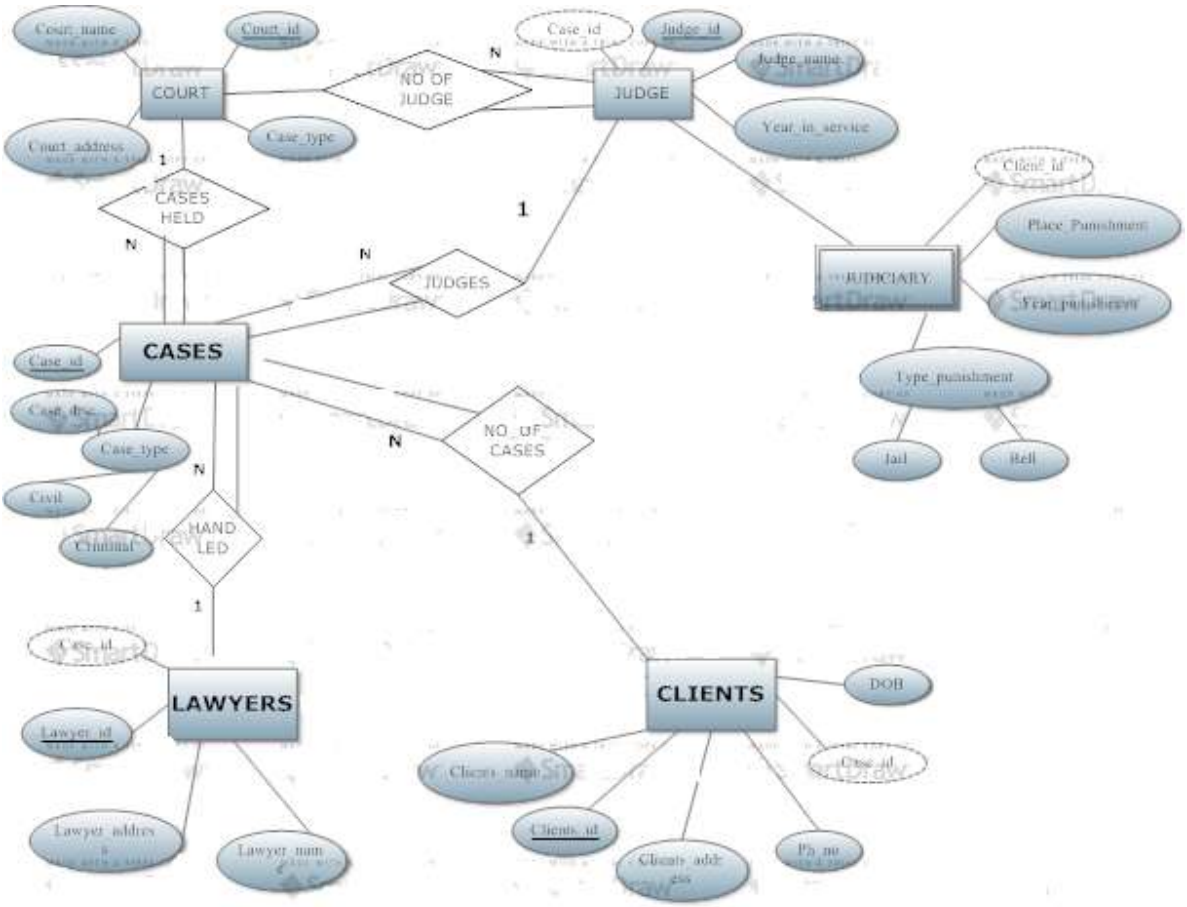
Server Software:Apache/2.4.9 (Win32) PHP/5.5.12

.

FEDERAL COURT DATABASE

DESIGN

ER DIAGRAM



MODULES

The various information systems are designed to be “model directed” in which mathematical models of court procedure are embedded in the systems so as to direct all system response to user input, and monitor the processing of cases relative to correct procedures and regulations.

MODULE DESCRIPTION

In order to increase case processing requirements within federal judiciary, Many major application areas were identified including management of criminal, civil, bankruptcy, and appellate cases, scheduling and calendaring, jury management, document processing, archival record maintenance, and electronic mail. Databases were planned and developed that would eventually hold millions of records within both national and distributed data centers.

The modules involved are :

- COURTS - Court is an entity. It has attributes like court_id which is primary attribute, the names of different courts, and the addresses of the different courts. The reference attribute used in this entity to relate the attribute cases is case_type.

- CASES - Case is an entity. It has attributes like case_id which is an primary attribute, description of the case, the type of the case it can either be civil or it can be criminal case.

- LAWYER - Lawyer is an entity. It has attributes like lawyer_id which is an primary attribute, lawyer_address along with lawyer_name. The reference attribute used in this entity to relate the attribute cases is case_id.

FEDERAL COURT DATABASE

- CLIENTS - Clients is an entity. It has attributes like client_id which is an primary attribute, the name of the client along with their addresses, phone_number and date of birth. The reference attribute used in this entity to relate the attribute cases is case_id.

- JUDGE - Judge is an entity. It has attributes like judge_id which is primary attribute, judge _name along with years in experience of the judge. The reference attribute used in this entity to relate the attribute case is case_id. Judge will be the admin in our project and admin has attributes like username and password

- JUDICIARY - Judiciary is an weak entity under the entity Judge. The attributes of this are type of punishment which has two attributes like jail and bail, years of punishment, place of punishment. In order to relate this weak entity judiciary with clients we have an weak relation with an reference attribute client_id.

CODE IMPLEMENTATION

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "court";
$conn = new mysqli($servername, $username, $password, $dbname); // Create connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
} // Check connection
$uname=$_POST["uname"];
$passw=$_POST["passw"];
$sql = "SELECT * FROM Admin where username='$uname' AND password='$passw'";//for check
$result = $conn->query($sql);
?>
<?php
if ($result->num_rows > 0) {
        echo 'Logged in successfully! <br>';
      while($row = $result->fetch_assoc()) {
      echo " - Name: " . $row["username"]. " Password: " . $row["password"]. "<br>";
```

```
        header('location:Adm.html');
    }// output data of each row


} else {
    echo "YOU ENTER WORONG Username AND Password PLEASE REENTER";
}
$conn->close();
?>
```

- Auto Increment

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "court";
$conn = new mysqli($servername, $username, $password, $dbname); // Create connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
} // Check connection
$Case_id=$_POST["Case_id"];
header('Location:client1_view.php?user='.$Client_id.");
?>
```

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "court";
 $usr_name=$_GET['user'];
$conn = new mysqli($servername, $username, $password, $dbname);// Create connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
} // Check connection
$sql1 = "SELECT Case_id,Case_description,Case_type,Case_date FROM cases where
Client_id='$usr_name' ";
$query_run=mysqli_query($conn,$sql1);
While($query_row=mysqli_fetch_array($query_run))
{
$Case_id=$query_row['Case_id'];
$Case_type=$query_row['Case_type'];
$Case_description=$query_row['Case_description'];
$Case_date=$query_row['Case_date'];
}
?
```

```
<html>
<h1>CASE DETAILS</h1>
                    <table border="1px">
                    <tr>
                    <th>Client_Fname</th>
                    <th>Client_Minit</th>
                    <th>Client_Lname</th>
                    <th>Client_address</th>
                    <th>Lawyer_name</th>
                    <th>Case_description_</th>
                        <th>Ph_no</th>
<th>DOB</th>
                <th>Case_type</th>
                <th>Case_date</th>
                </tr>
                <tr>
                    <td><?php echo $Client_Fname; ?></td>
                    <td><?php echo $Client_Minit; ?></td>
                    <td><?php echo $Client_Lname; ?></td>
                    <td><?php echo $Client_address; ?></td>
```

```
<td><?php echo $Lawyer_name; ?></td>

<td><?php echo $Case_description; ?></td>

<td><?php echo $Ph_no; ?></td>

<td><?php echo $DOB; ?></td>

<td><?php echo $Case_type; ?></td>

<td><?php echo $Case_date; ?></td>

</tr>

</table

</html>
```

SNAPSHOTS

- FORMS:

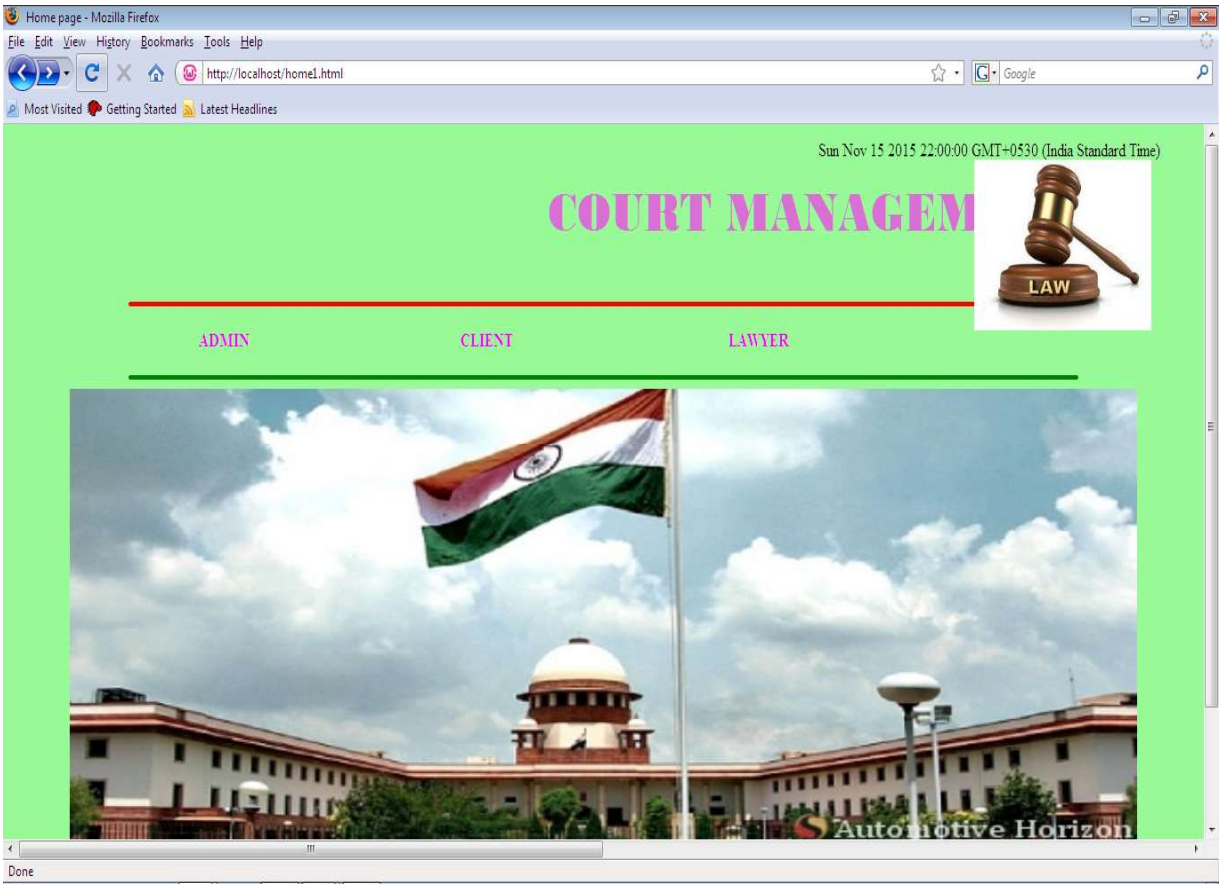


Fig 1. Homepage

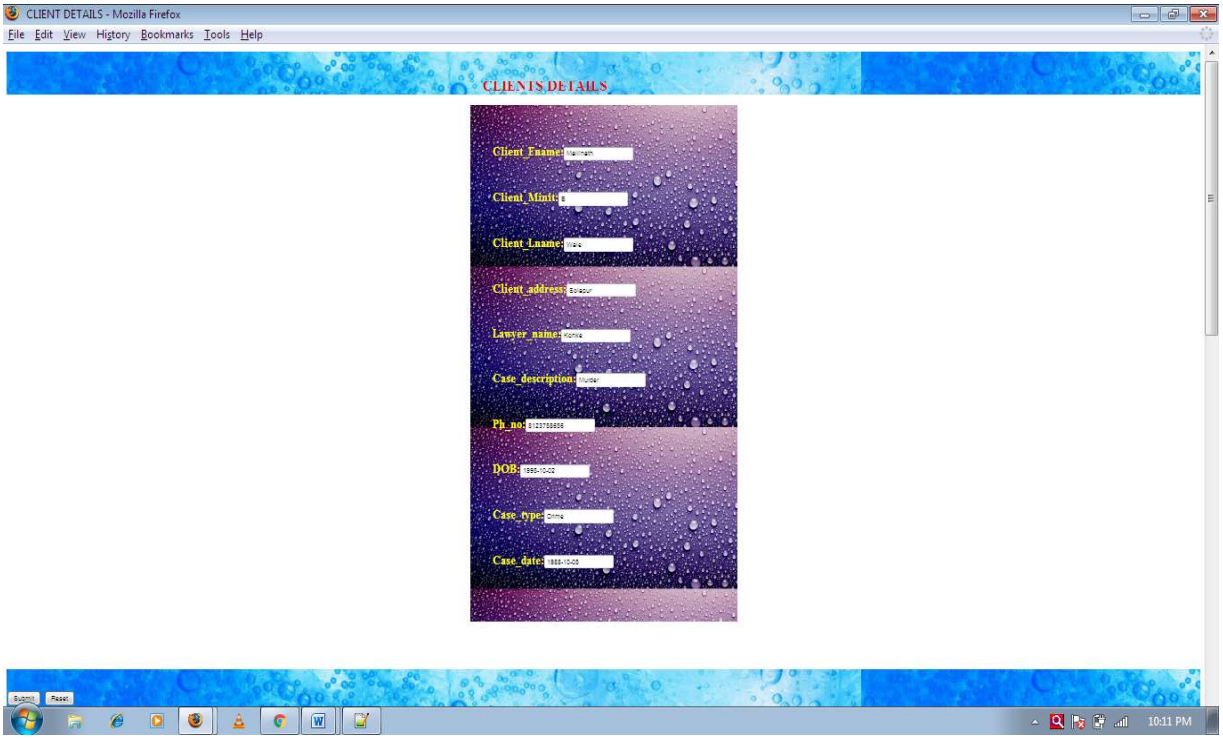
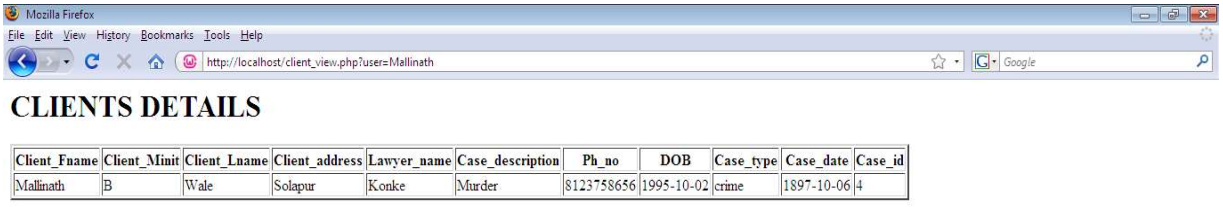


Fig 2. Client Details



CLIENTS DETAILS

Client_Fname	Client_Minit	Client_Lname	Client_address	Lawyer_name	Case_description	Ph_no	DOB	Case_type	Case_date	Case_id
Mallinath	B	Wale	Solapur	Konke	Murder	8123758656	1995-10-02	crime	1897-10-06	4

Fig 3. Client Details Are Stored.

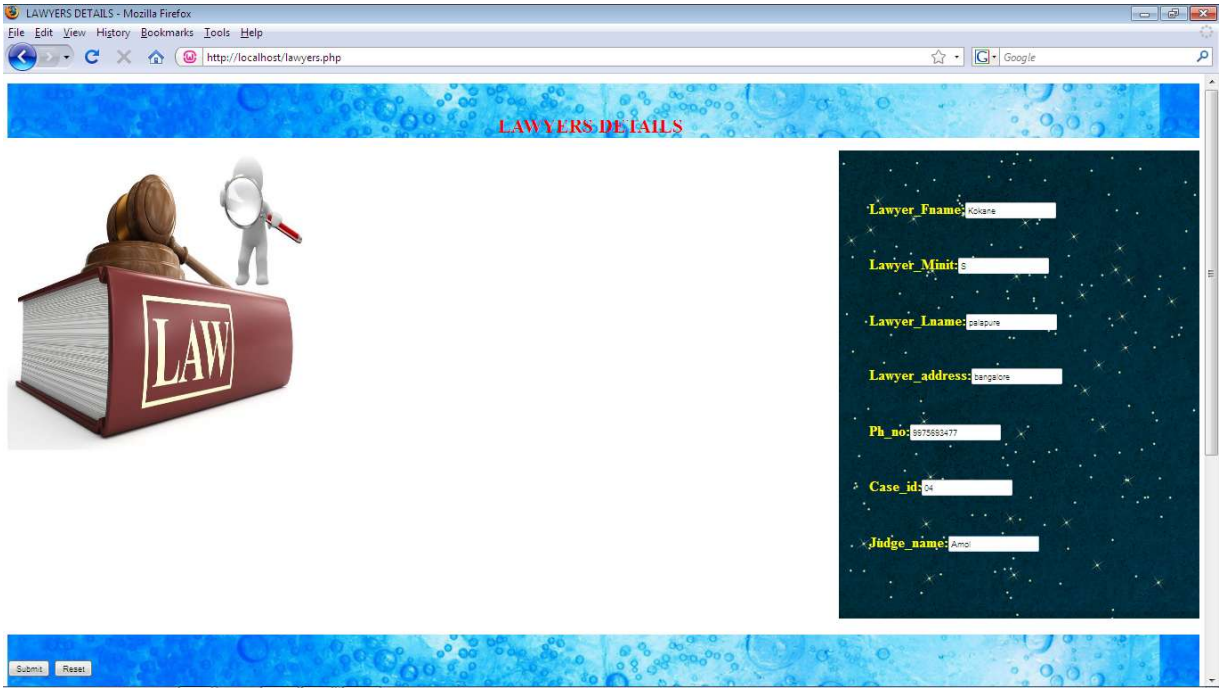


Fig 4. Lawyer Details.

Fig 5. New Lawyer Details Are Stored.

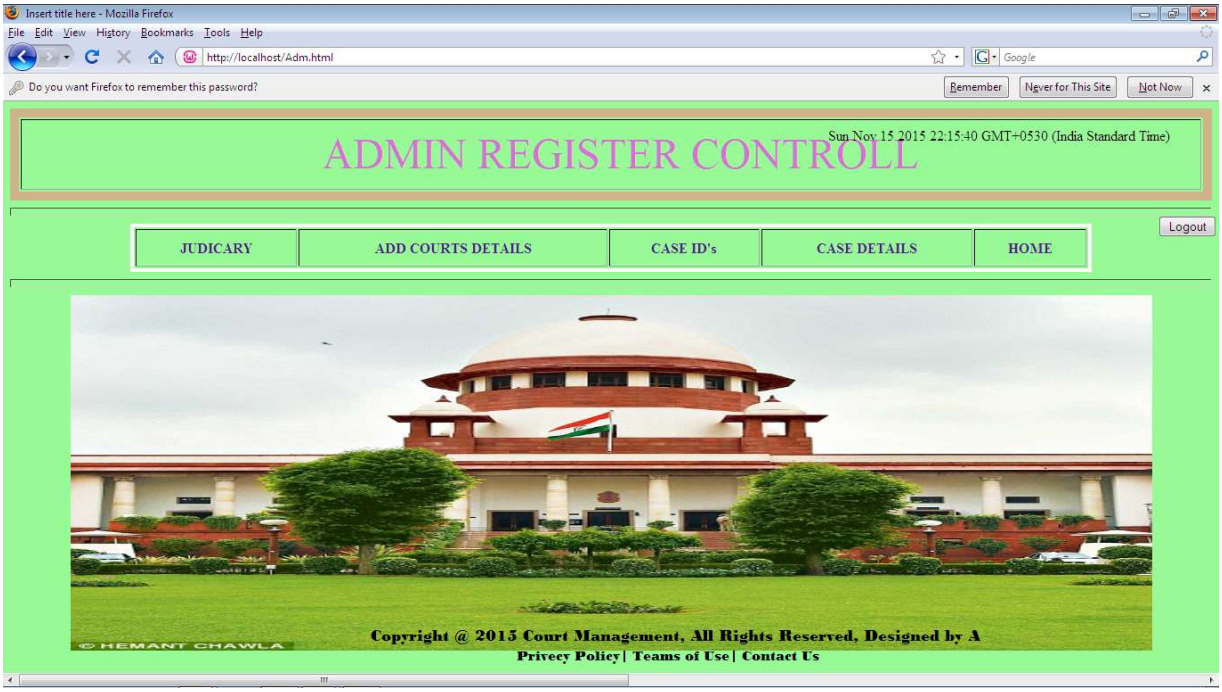


Fig 6. Admin Page (Judge)

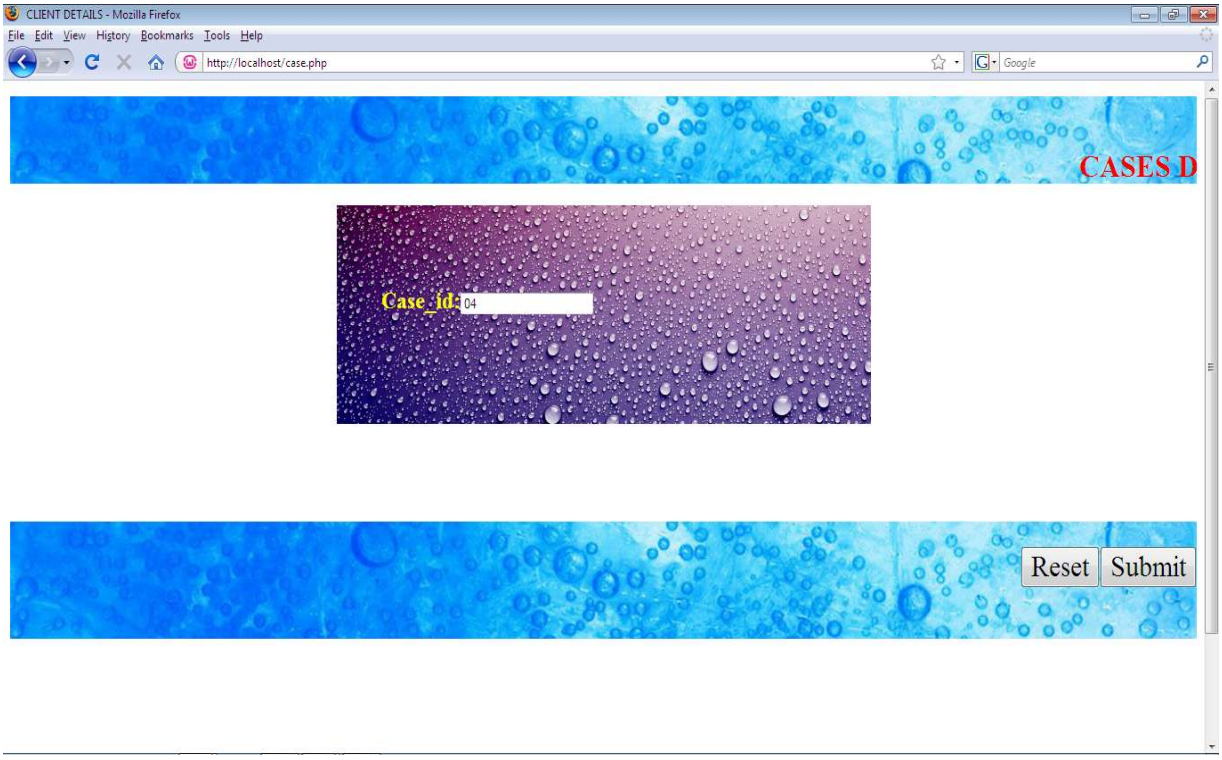
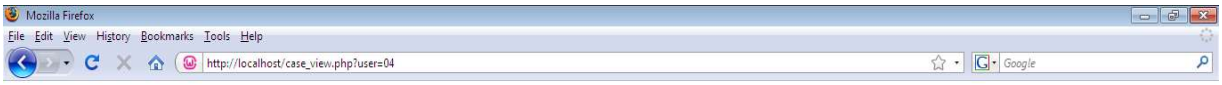


Fig 7. Form To View Case Details.



Results

CASE DETAILS

Client_Fname	Client_Minit	Client_Lname	Client_address	Lawyer_name	Case_description_	Ph_no	DOB	Case_type	Case_date
Mallinath	B	Wale	Solapur	Konke	Murder	8123758656	1995-10-02	crime	1897-10-06

Fig 8. Case Details Viewed.

Conclusion

This project is effective to store and retrieve information about different cases held in a court. In future this project “FEDERAL COURT DATABASE” can be enhanced and made more effective by making one particular database to all the courts in a particular region or an state.

References

1. Website- www.w3schools.com.
2. Professional PHP Language-k'evin Hough.
3. My SQL Tutorial.