

# Department of CSE

## SSN College of Engineering

Vishakan Subramanian - 18 5001 196 - Semester VI

30 March 2021

---

### UCS 1611 - Internet Programming Lab

---

#### Exercise 4: Java Servlets and MySQL

#### Learning Objective:

Create a HTML file with following form fields:

1. Employee ID
2. Employee Name
3. Designation
4. Department
5. Basic Salary
6. Phone Number
7. Address
8. Date of Birth
9. Gender
10. Marital Status
11. Submit (Button, onclick submit the form details to a servlet)

## 12. Reset (Button, onclick)

Write a servlet program which reads all form details, finds the gross pay based on basic pay and designation. The servlet returns all details (a-j, gross pay) in table format to the client.

Develop a Patient Management System (PMS) in a Hospital that maintains the patient records, with the following specifications:

1. Admin only can access the patient records.
2. Admin has to authenticate himself with the credentials to access the patient details.
3. The first page is the login page (login.html) where the admin should enter the credentials.
4. If login fails, display an alert "Login Failed, enter the correct details", clear the username and password textbox and focus the control back to the username of login.html.
5. Perform the authentication using JavaScript, use hardcoded username and password in login.html.
6. On successful login, it should be redirected to homepage (homepage.html) of PMS.
7. The homepage.html should display the following: Add Record, Update Record, Delete Record, View Record & Search Record.
8. Clicking the links should invoke its corresponding html file from where the information is submitted to the corresponding Servlet.
9. When "View" button is clicked it invokes the "ViewServlet" to display all the records in the database.
10. Similarly, perform ADD, UPDATE, VIEW and SEARCH using corresponding html and Servlet.
11. UPDATE is invoked if phone number or Address of the patient is changed.
12. SEARCH for particular patient based on Patient ID.
13. After each operation ViewServlet should be invoked to view the changes in the Database.

Database Name: Patient\_Details

Table Schema: Name, Age, ID, Gender, Address, Phone Number, Marital status, Date of Visit.

## Code - Employee Registration Page HTML:

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <meta charset="utf-8">
6     <meta name="viewport" content="width=device-width">
7     <title>Reg. Form</title>
8     <link rel="stylesheet" href="styles.css">
9 </head>
10
11 <body>
12     <h1 class="titleText">Employee Registration Form</h1>
13     <hr>
14     <br><br>
15
16     <div class="reg-page">
17         <form action="Form" name="regform" id="regform" method="GET">
18             <label for="eid">Employee ID</label><br>
19             <input type="text" name="eid" placeholder="SSN0203" size="30"
required>
20             <br><br>
21
22             <label for="ename">Employee Name</label><br>
23             <input type="text" name="ename" placeholder="Dwight Schrute"
size="30" required title="Enter English alphabets only">
24             <br><br>
25
26             <label for="job">Designation</label><br>
27             <input type="text" name="job" placeholder="AARM" size="30"
required>
28             <br><br>
29
30             <label for="dept">Department</label><br>
31             <input type="text" name="dept" placeholder="Sales" size="30"
required>
32             <br><br>
33
34             <label for="basic">Basic Salary</label><br>
35             <input type="text" name="basic" placeholder="40000" size="30"
pattern="^\d+(\.\d{1,2})? $" required>
36             <br><br>
37
38             <label for="phone">Phone Number</label><br>
39             <input type="text" name="phone" placeholder="( +91)
123-456-7890" size="20">
40             <br><br>
41
```

```

42     <label for="addr">Address</label><br>
43     <textarea cols="30" rows="3" name="addr" required></textarea>
44     <br><br>
45
46     <label for="dob">Date of Birth</label><br>
47     <input type="date" name="dob" required><br><br>
48
49     <label>Gender</label>
50     <br>
51     <input type="radio" id="male" name="gender" value="Male"
required>
52     <label for="male">Male</label>
53     <input type="radio" id="female" name="gender" value="Female"
required>
54     <label for="female">Female</label>
55     <input type="radio" id="other" name="gender" value="Other"
required>
56     <label for="other">Other</label>
57
58     <br><br>
59
60     <label for="marital">Marital Status</label><br>
61     <select name="marital" required>
62         <option>Single</option>
63         <option>In A Relationship</option>
64         <option>Married</option>
65         <option>Separated</option>
66         <option>Divorced</option>
67         <option>Widowed</option>
68     </select>
69     <br><br>
70
71     <input type="submit" class="btn" name="Submit" id="Submit"
value="Register"></input>
72     <input type="reset" class="btn" name="Reset" id="Reset" value=
"Clear All"></input>
73     </form>
74 </div>
75
76 <footer>
77     &copy; Vishakan Subramanian 2021
78 </footer>
79 </body>
80
81 </html>

```

## Code - Employee Registration Page CSS:

```
1 @import url("https://fonts.googleapis.com/css2?family=Barlow+Semi+
   Condensed&display=swap");
2
3 body {
4     background: #76b852;
5     background: linear-gradient(to left, #76b852, #8dc26f);
6     font-family: "Barlow Semi Condensed", -apple-system, "Segoe UI", sans-
       serif;
7 }
8
9 .titleText {
10     text-align: center;
11 }
12
13 .reg-page {
14     width: 500px;
15     position: relative;
16     z-index: 1;
17     background: #ffffff;
18     max-width: 500px;
19     margin: 0 auto 100px;
20     padding: 45px;
21     text-align: left;
22     box-shadow: 0 0 20px 0 rgba(0, 0, 0, 0.2), 0 5px 5px 0 rgba(0, 0, 0,
       0.24);
23 }
24
25 .reg-page input,
26 .reg-page textarea,
27 .reg-page select {
28     outline: 0;
29     background: #f2f2f2;
30     width: 100%;
31     border: 0;
32     margin: 15px 0px 15px 40px;
33     padding: 15px;
34     box-sizing: border-box;
35     font-size: 16px;
36     font-family: "Barlow Semi Condensed", sans-serif;
37 }
38
39 .reg-page label {
40     font-weight: 650;
41 }
42
43 .reg-page textarea {
44     resize: none;
```

```

45 }
46
47 #Submit,
48 #Reset{
49     font-family: "Barlow Semi Condensed", sans-serif;
50     font-size: 16px;
51     font-weight: 300;
52     text-transform: uppercase;
53     outline: 0;
54     background: #4caf50;
55     width: 200px;
56     border: 0;
57     padding: 15px;
58     color: #ffffff;
59     transition: all 0.3 ease;
60     cursor: pointer;
61     user-select: none;
62 }
63
64
65 #Submit:hover, #Reset:hover,
66 #Submit:active, #Reset:active,
67 #Submit:focus, #Reset:focus{
68     font-weight: bold;
69     background: #43a047;
70 }
71
72 footer {
73     color: black;
74     font-weight: bold;
75     border-top: 3px solid black;
76     padding: 5px 5px 5px 5px;
77     text-align: center;
78     width: 100%;
79 }
80
81 table, td, th{
82     font-size: 25px;
83     border: 2px solid black;
84     border-collapse: collapse;
85     text-align: center;
86 }
87
88 table{
89     width: 75%;
90 }
91
92 tr:hover{
93     background-color: azure;
94 }

```

## Code - Web XML:

```
1 <web-app>
2     <servlet>
3         <servlet-name>Form</servlet-name>
4         <servlet-class>Form</servlet-class>
5     </servlet>
6
7     <servlet-mapping>
8         <servlet-name>Form</servlet-name>
9         <url-pattern>/Form</url-pattern>
10    </servlet-mapping>
11
12    <welcome-file-list>
13        <welcome-file>index.html</welcome-file>
14    </welcome-file-list>
15 </web-app>
```

## Code - Employee Servlet:

```
1 import javax.servlet.http.*;
2 import javax.servlet.*;
3 import javax.servlet.ServletException;
4 import java.io.*;
5
6 public class Form extends HttpServlet{
7
8     public double getNetPay(double basic){
9         //to calculate the net pay of the employee
10        double da, hra, pf, tax, gross, net;
11        da = 0.30 * basic;
12        hra = 0.15 * basic;
13        pf = 0.05 * basic;
14        tax = 0.10 * basic;
15        gross = basic + da + hra;
16        net = gross - pf - tax;
17
18        return net;
19    }
20
21    public void doGet(HttpServletRequest req, HttpServletResponse res)
22    throws ServletException, IOException{
23
24        res.setContentType("text/html; charset=UTF-8"); //defining action
for GET
25
26        PrintWriter out = res.getWriter(); //get the stream
to write the data
27
28        try{
29            String eid = req.getParameter("eid").trim();
30            String ename = req.getParameter("ename").trim();
31            String job = req.getParameter("job").trim();
32            String dept = req.getParameter("dept").trim();
33            double basic = Double.parseDouble(req.getParameter("basic").
trim());
34            String phone = req.getParameter("phone").trim();
35            String addr = req.getParameter("addr").trim();
36            String dob = req.getParameter("dob").trim();
37            String gender = req.getParameter("gender").trim();
38            String mstatus = req.getParameter("marital").trim();
39
40            double net = getNetPay(basic);
41            String netStr = String.format("%.2f", net);
42
43            out.println("<!DOCTYPE html><html>");
44            out.println("<head><title>Employee Details</title>");
```



```

45         out.println("<link rel=\"stylesheet\" href=\"styles.css\"></head>");
46         out.println("<body><h1 class=\"titleText\">Employee Details</h1><hr><br>");
47         out.println("<table align=\"center\"><tr><th>Employee ID</th><td>" + eid + "</td></tr>");
48         out.println("<tr><th>Employee Name</th><td>" + ename + "</td></tr>");
49         out.println("<tr><th>Date of Birth</th><td>" + dob + "</td></tr>");
50         out.println("<tr><th>Gender</th><td>" + gender + "</td></tr>");
51         out.println("<tr><th>Marital Status</th><td>" + mstatus + "</td></tr>");
52         out.println("<tr><th>Designation</th><td>" + job + "</td></tr>");
53         out.println("<tr><th>Department</th><td>" + dept + "</td></tr>");
54         out.println("<tr><th>Address</th><td>" + addr + "</td></tr>");
55         out.println("<tr><th>Phone</th><td>" + phone + "</td></tr>");
56         out.println("<tr><th>Basic Pay</th><td>" + basic + "</td></tr>");
57         out.println("<tr><th>Net Pay</th><td>" + netStr + "</td></tr>");
58         out.println("</table></body>");
59         out.println("</html>");
60     }
61 }
62
63 catch(Exception e){
64     out.println(e.getMessage());
65 }
66
67 finally{
68     out.close();
69 }
70 }
71 }

```

## Output - Employee Registration Page:

Figure 1: Browser Output: Employee Registration Page.

### Employee Registration Form

Employee ID

SSN0203

Employee Name

Dwight Schrute

Designation

AARM

Department

Sales

Basic Salary

40000

Phone Number

9694203450

Address

34, Sunset Blvd, Scranton, PA

Date of Birth

16/06/1967

Gender

☒ Male ☐ Female ☐ Other

Marital Status

Married

REGISTER

CLEAR ALL

© Vishakan Subramanian 2021

## Output - Employee Servlet Page:

Figure 2: Browser Output: Employee Servlet Page.

Employee Details	
Employee ID	SSN0203
Employee Name	Dwight Schrute
Date of Birth	1967-06-16
Gender	Male
Marital Status	Married
Designation	AARM
Department	Sales
Address	34, Sunset Blvd, Scranton, PA
Phone	9694203450
Basic Pay	40000.0
Net Pay	52000.00

## Code - Login Page HTML:

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <title>PMS Portal</title>
6     <link rel="stylesheet" href="css/pages.css">
7     <script src="scripts/login.js"></script>
8 </head>
9
10 <body>
11     <h1 class="titleText">Welcome to Patient Mangement System</h1>
12     <hr>
13     <h2 class="titleText">Login Portal</h2><br><br>
14     <div align="center" class="reg-page">
15         <label for="id">Username:</label>
16         <input type="text" name="uname" id="uname" /><br><br>
17         <label for="id">Password:</label>
18         <input type="password" name="pwd" id="pwd" /><br><br>
19         <input type="submit" value="Login" id="button" class="btn" onclick
            ="validateUser()" />
20     </div>
21
22 </body>
23
24 </html>
```

## Code - Home Page HTML:

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <title>PMS Portal</title>
6     <link rel="stylesheet" href="css/pages.css">
7 </head>
8
9 <body>
10     <h1 class="titleText">Patient Management System</h1>
11     <hr>
12     <h2 class="titleText">Navigation Menu</h2><br><br>
13     <div align="center" class="reg-page">
14         <button class="btn"><a href="add.html", target="_blank" class="
links">Add A Record</a></button><br><br>
15         <button class="btn"><a href="update.html", target="_blank" class="
links">Update A Record</a></button><br><br>
16         <button class="btn"><a href="delete.html", target="_blank" class="
links">Delete A Record</a></button><br><br>
17         <button class="btn" onclick="clearAction()"><a href="view.html",
target="_blank" class="links">View All Records</a></button><br><br>
18         <button class="btn"><a href="search.html", target="_blank" class="
links">Search A Record</a></button><br><br>
19     </div>
20
21     <script>
22         clearAction = () =>{
23             localStorage.setItem("action", ""); //clearing the action
field for the view page
24         }
25     </script>
26
27 </body>
28
29 </html>
```

## Code - Add Record Page HTML:

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <meta charset="utf-8">
6     <meta name="viewport" content="width=device-width">
7     <title>PMS - Add Record</title>
8     <link href="css/pages.css" rel="stylesheet" type="text/css" />
9 </head>
10
11 <body>
12     <h1 class="titleText">Patient Management System</h1>
13     <hr>
14     <h2 class="titleText">Add A Record</h2>
15     <br><br>
16
17     <div class="reg-page">
18         <form name="regform" id="regform" action="AddRecord" method="GET">
19             <label for="id">Patient ID</label><br>
20             <input type="text" id="id" name="id" placeholder="P2008" size=
"10" required pattern="^P[0-9]+$" title="Enter in the specified format"
>
21             <br><br>
22
23             <label for="pname">Name</label><br>
24             <input type="text" id="pname" name="pname" placeholder="John
Doe" size="40" required pattern="^[a-zA-Z\s]+$" title="Enter English
alphabets only">
25             <br><br>
26
27             <label for="age">Age</label><br>
28             <input type="number" id="age" name="age" min="1" max="120"
step="1" placeholder="21" required>
29             <br><br>
30
31             <label>Gender</label>
32             <br>
33             <input type="radio" id="male" name="gender" value="Male"
required>
34             <label for="male">Male</label>
35             <input type="radio" id="female" name="gender" value="Female"
required>
36             <label for="female">Female</label>
37             <input type="radio" id="other" name="gender" value="Other"
required>
38             <label for="other">Other</label>
39
```

```

40
41     <br><br>
42
43     <label for="marital">Marital Status</label><br>
44     <select id="marital" name="marital" required>
45         <option>Single</option>
46         <option>In A Relationship</option>
47         <option>Married</option>
48         <option>Separated</option>
49         <option>Divorced</option>
50         <option>Widowed</option>
51     </select>
52     <br><br>
53
54     <label for="addr">Address</label><br>
55     <textarea cols="30" rows="3" id="addr" name="addr" required></
textarea>
56     <br><br>
57
58     <label for="phone">Phone Number</label><br>
59     <input type="text" id="phone" name="phone" placeholder="
1234567890" size="20" required>
60     <br><br>
61
62     <label for="dov">Date of Visit</label><br>
63     <input type="date" id="dov" name="dov" required><br><br>
64
65     <input type="submit" class="btn" name="Submit" id="Submit"
value="Register"></input><br>
66     <input type="reset" class="btn" name="Reset" id="Reset" value=
"Clear All"></input>
67     </form>
68 </div>
69
70 <script>
71     localStorage.setItem("action", "Record Added Successfully!");
72 </script>
73
74 </body>
75
76 </html>

```

## Code - Update Record Page HTML:

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <meta charset="utf-8">
6     <meta name="viewport" content="width=device-width">
7     <title>PMS - Update Record</title>
8     <link href="css/pages.css" rel="stylesheet" type="text/css" />
9 </head>
10
11 <body>
12     <h1 class="titleText">Patient Management System</h1><hr>
13     <h2 class="titleText">Update A Record</h2><br><br>
14
15     <div class="reg-page">
16         <form name="regform" id="regform" method="GET" action="
UpdateRecord">
17             <label for="id">Patient ID</label><br>
18             <input type="text" id="id" name="id" placeholder="P2008" size=
"10" required pattern="^P[0-9]+$" title="Enter in the specified format"
>
19
20             <br><br>
21
22             <label for="addr">Updated Address</label><br>
23             <textarea cols="30" rows="3" id="addr" name="addr" required></
textarea>
24
25             <br><br>
26
27             <label for="phone">Updated Phone Number</label><br>
28             <input type="text" id="phone" name="phone" placeholder="
1234567890" size="20" required>
29
30             <br><br>
31             <input type="submit" class="btn" name="Submit" id="Submit"
value="Update"></input><br>
32             <input type="reset" class="btn" name="Reset" id="Reset" value=
"Clear All"></input>
33         </form>
34     </div>
35
36     <script>
37         localStorage.setItem("action", "Record Updated Successfully!");
38     </script>
39 </body>
40 </html>
```



## Code - Delete Record Page HTML:

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <title>PMS - Delete Record</title>
6     <link rel="stylesheet" href="css/pages.css">
7 </head>
8
9 <body>
10     <h1 class="titleText">Patient Management System</h1>
11     <hr>
12     <h2 class="titleText">Delete A Record</h2><br><br>
13     <div class="reg-page">
14         <form name="form1" method="GET" action="DeleteRecord">
15             <label for="id">Patient ID:</label><br>
16             <input type="text" name="pid" /><br><br>
17             <input type="submit" class="btn" value="Delete Record" id="
button" />
18         </form>
19     </div>
20
21     <script>
22         localStorage.setItem("action", "Record Deleted Successfully!");
23     </script>
24
25 </body>
26
27 </html>
```

## Code - View Records Page HTML:

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <title>PMS - View Records</title>
6     <link rel="stylesheet" href="css/pages.css">
7 </head>
8
9 <body>
10     <h1 class="titleText">Patient Management System</h1>
11     <hr>
12     <h2 class="titleText">View All Records</h2><br><br>
13     <h3 class="titleText" id="responseText"></h3>
14     <div class="reg-page">
15         <form name="form2" method="GET" action="ViewRecords">
16             <input type="submit" class="btn" value="View" id="button" />
17         </form>
18     </div>
19
20     <script>
21         document.getElementById("responseText").innerHTML = localStorage.
getItem("action");
22     </script>
23
24 </body>
25
26 </html>
```

## Code - Search Records Page HTML:

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <title>PMS - Search Record</title>
6     <link rel="stylesheet" href="css/pages.css">
7 </head>
8
9 <body>
10     <h1 class="titleText">Patient Management System</h1>
11     <hr>
12     <h2 class="titleText">Search A Record</h2><br><br>
13     <div class="reg-page">
14         <form name="form2" method="GET" action="SearchRecord">
15             <label for="id">Patient ID:</label><br>
16             <input type="text" name="pid" /><br><br>
17             <input type="submit" class="btn" value="Search Record" id="
button" />
18         </form>
19     </div>
20
21 </body>
22
23 </html>
```

## Code - Pages CSS:

```
1 @import url("https://fonts.googleapis.com/css2?family=Barlow+Semi+
   Condensed&display=swap");
2
3 body {
4     background: #76b852;
5     background: linear-gradient(to left, #76b852, #8dc26f);
6     font-family: "Barlow Semi Condensed", -apple-system, "Segoe UI", sans-
       serif;
7 }
8
9 .titleText,
10 .responseText {
11     text-align: center;
12 }
13
14 .reg-page {
15     width: 500px;
16     position: relative;
17     z-index: 1;
18     background: #ffffff;
19     max-width: 500px;
20     margin: 0 auto 100px;
21     padding: 45px;
22     text-align: left;
23     box-shadow: 0 0 20px 0 rgba(0, 0, 0, 0.2), 0 5px 5px 0 rgba(0, 0, 0,
       0.24);
24 }
25
26 .reg-page input,
27 .reg-page textarea,
28 .reg-page select {
29     outline: 0;
30     background: #f2f2f2;
31     width: 100%;
32     border: 0;
33     margin: 15px 0px 15px 40px;
34     padding: 15px;
35     box-sizing: border-box;
36     font-size: 16px;
37     font-family: "Barlow Semi Condensed", sans-serif;
38 }
39
40 .reg-page label {
41     font-weight: 650;
42 }
43
44 .reg-page textarea {
```

```

45     resize: none;
46 }
47
48 .reg-page .btn {
49     margin-left: 140px;
50     font-family: "Barlow Semi Condensed", sans-serif;
51     font-size: 16px;
52     font-weight: 300;
53     text-transform: uppercase;
54     outline: 0;
55     background: #4caf50;
56     width: 200px;
57     border: 0;
58     padding: 15px;
59     color: #ffffff;
60     transition: all 0.3 ease;
61     cursor: pointer;
62     user-select: none;
63 }
64
65 .btn:hover,
66 .btn:focus,
67 .btn:active {
68     font-weight: bold;
69     background: #43a047;
70 }
71
72 footer {
73     color: black;
74     font-weight: bold;
75     border-top: 3px solid black;
76     padding: 5px 5px 5px 5px;
77     text-align: center;
78     width: 100%;
79 }
80
81 #addict-list li {
82     display: inline-block;
83     appearance: button;
84     padding: 2px 5px;
85     background-color: #76b852;
86     border-radius: 25px;
87 }
88
89 table,
90 td,
91 th {
92     font-size: 25px;
93     border: 2px solid black;
94     padding: 5px;
95     border-collapse: collapse;

```

```
96     text-align: center;
97 }
98
99 table {
100     width: 75%;
101 }
102
103 tr:hover {
104     background-color: azure;
105 }
106
107 .links{
108     text-decoration: none;
109     cursor: pointer;
110     color: white;
111 }
112
113 .links:hover{
114     font-weight: bold;
115 }
```

## Code - Script JS:

```
1 function validateUser() {  
2     //To validate the login of the administrator  
3  
4     var user = document.getElementById("uname").value;  
5     var pass = document.getElementById("pwd").value;  
6  
7     if (user === "tomcat" && pass === "root") {  
8         document.getElementById("uname").value = "";  
9         document.getElementById("pwd").value = "";  
10        window.open("homepage.html", target="_self");  
11    }  
12    else {  
13        alert("Invalid credentials. Kindly check and try again.");  
14        document.getElementById("uname").value = "";  
15        document.getElementById("pwd").value = "";  
16        document.getElementById("uname").focus();  
17    }  
18 }
```

## Code - Script SQL:

```
1 USE Patient_Details;
2
3 /* Records Table Creation */
4
5 DROP TABLE Records;
6
7 CREATE TABLE Records(
8     pid      CHAR(5) PRIMARY KEY,
9     pname    VARCHAR(40),
10    age       NUMERIC(3),
11    gender    CHAR(1),
12    mstatus   VARCHAR(25),
13    addr      VARCHAR(100),
14    phone     NUMERIC(12),
15    dov       DATE
16 );
17
18 /* Inserting Dummy Records */
19
20 INSERT INTO Records VALUES(
21     "P2001",
22     "Chris Paley",
23     25,
24     "M",
25     "Single",
26     "275, Sunset Blvd, Los Angeles, CA",
27     9887521131,
28     "2021-01-21"
29 );
30
31 INSERT INTO Records VALUES(
32     "P2002",
33     "Rick Riordan",
34     45,
35     "M",
36     "Married",
37     "275, Olympus Avenue, Greek County, NJ",
38     9444421212,
39     "2021-02-23"
40 );
```



## Code - Web XML:

```
1 <web-app>
2   <servlet>
3     <servlet-name>AddRecord</servlet-name>
4     <servlet-class>AddRecord</servlet-class>
5   </servlet>
6
7   <servlet>
8     <servlet-name>DeleteRecord</servlet-name>
9     <servlet-class>DeleteRecord</servlet-class>
10  </servlet>
11
12  <servlet>
13    <servlet-name>UpdateRecord</servlet-name>
14    <servlet-class>UpdateRecord</servlet-class>
15  </servlet>
16
17  <servlet>
18    <servlet-name>SearchRecord</servlet-name>
19    <servlet-class>SearchRecord</servlet-class>
20  </servlet>
21
22  <servlet>
23    <servlet-name>ViewRecords</servlet-name>
24    <servlet-class>ViewRecords</servlet-class>
25  </servlet>
26
27  <servlet-mapping>
28    <servlet-name>AddRecord</servlet-name>
29    <url-pattern>/AddRecord</url-pattern>
30  </servlet-mapping>
31
32  <servlet-mapping>
33    <servlet-name>DeleteRecord</servlet-name>
34    <url-pattern>/DeleteRecord</url-pattern>
35  </servlet-mapping>
36
37  <servlet-mapping>
38    <servlet-name>UpdateRecord</servlet-name>
39    <url-pattern>/UpdateRecord</url-pattern>
40  </servlet-mapping>
41
42  <servlet-mapping>
43    <servlet-name>SearchRecord</servlet-name>
44    <url-pattern>/SearchRecord</url-pattern>
45  </servlet-mapping>
46
47  <servlet-mapping>
```

```
48         <servlet-name>ViewRecords</servlet-name>
49         <url-pattern>/ViewRecords</url-pattern>
50     </servlet-mapping>
51
52     <welcome-file-list>
53         <welcome-file>login.html</welcome-file>
54     </welcome-file-list>
55 </web-app>
```

## Code - Add Record Servlet:

```
1 import java.io.*;
2 import java.sql.*;
3 import java.util.*;
4 import javax.servlet.*;
5 import javax.servlet.http.*;
6
7 public class AddRecord extends HttpServlet{
8     public void doGet(HttpServletRequest request, HttpServletResponse
9         response)
10         throws ServletException, IOException{
11
12         //DB_URL: jdbc:mysql:location:port/DatabaseName?Parameters
13
14         String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
15         String DB_URL = "jdbc:mysql://localhost:3306/Patient_Details?
16         autoReconnect=true&useSSL=false";
17         String USER = "admin";
18         String PASS = "password";
19
20         PrintWriter out = response.getWriter();
21
22         String pid = request.getParameter("id");
23         String name = request.getParameter("pname");
24         int age = Integer.parseInt(request.getParameter("age"));
25         String gdr = request.getParameter("gender");
26
27         switch(gdr){
28             case "Male":
29                 gdr = "M";
30                 break;
31             case "Female":
32                 gdr = "F";
33                 break;
34             case "Other":
35                 gdr = "O";
36                 break;
37             default:
38                 gdr = null;
39         }
40
41         String mstatus = request.getParameter("marital");
42         String addr = request.getParameter("addr");
43         long phone = Long.parseLong(request.getParameter("phone"));
44         String dov = request.getParameter("dov");
45
46         try{
47             Class.forName(JDBC_DRIVER);
```

```

46         Connection conn = DriverManager.getConnection(DB_URL, USER,
PASS);
47
48         String insertCmd = "INSERT INTO Records VALUES(?, ?, ?, ?, ?,
?, ?, ?)";
49         PreparedStatement stmt = conn.prepareStatement(insertCmd);
50         stmt.setString(1, pid);
51         stmt.setString(2, name);
52         stmt.setInt(3, age);
53         stmt.setString(4, gdr);
54         stmt.setString(5, mstatus);
55         stmt.setString(6, addr);
56         stmt.setLong(7, phone);
57         stmt.setString(8, dov);
58
59         stmt.executeUpdate();    //execute the PreparedStatement
60
61         //To open the view.html page after adding the record
62         RequestDispatcher rd = request.getRequestDispatcher("/view.
html");
63         rd.forward(request, response);
64
65         stmt.close();
66         conn.close();
67     }
68
69     catch(SQLException se){
70         se.printStackTrace();
71         out.println(se);
72     }
73
74     catch(Exception e){
75         e.printStackTrace();
76         out.println(e);
77     }
78 }
79 }

```

## Code - Update Record Servlet:

```
1 import java.io.*;
2 import java.sql.*;
3 import java.util.*;
4 import javax.servlet.*;
5 import javax.servlet.http.*;
6
7 public class UpdateRecord extends HttpServlet{
8     public void doGet(HttpServletRequest request, HttpServletResponse
9         response)
10         throws ServletException, IOException{
11         //DB_URL: jdbc:mysql:location:port/DatabaseName?Parameters
12
13         String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
14         String DB_URL = "jdbc:mysql://localhost:3306/Patient_Details?
15         autoReconnect=true&useSSL=false";
16         String USER = "admin";
17         String PASS = "password";
18
19         PrintWriter out = response.getWriter();
20
21         String pid = request.getParameter("id");
22         String addr = request.getParameter("addr");
23         long phone = Long.parseLong(request.getParameter("phone"));
24
25         try{
26             Class.forName(JDBC_DRIVER);
27             Connection conn = DriverManager.getConnection(DB_URL, USER,
28                 PASS);
29
30             String updateCmd = "UPDATE Records SET addr = ?, phone = ?
31             WHERE pid = ?";
32             PreparedStatement stmt = conn.prepareStatement(updateCmd);
33             stmt.setString(1, addr);
34             stmt.setLong(2, phone);
35             stmt.setString(3, pid);
36             stmt.executeUpdate(); //execute the PreparedStatement
37
38             //To open the view.html page after adding the record
39             RequestDispatcher rd = request.getRequestDispatcher("/view.
40             html");
41             rd.forward(request, response);
42
43             stmt.close();
44             conn.close();
45         }
46     }
```

```
43         catch(SQLException se){
44             se.printStackTrace();
45             out.println(se);
46         }
47
48         catch(Exception e){
49             e.printStackTrace();
50             out.println(e);
51         }
52     }
53 }
```

## Code - Delete Record Servlet:

```
1 import java.io.*;
2 import java.sql.*;
3 import java.util.*;
4 import javax.servlet.*;
5 import javax.servlet.http.*;
6
7 public class DeleteRecord extends HttpServlet{
8     public void doGet(HttpServletRequest request, HttpServletResponse
9         response)
10         throws ServletException, IOException{
11
12         //DB_URL: jdbc:mysql:location:port/DatabaseName?Parameters
13
14         String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
15         String DB_URL = "jdbc:mysql://localhost:3306/Patient_Details?
16         autoReconnect=true&useSSL=false";
17         String USER = "admin";
18         String PASS = "password";
19
20         PrintWriter out = response.getWriter();
21
22         String pid = request.getParameter("pid");
23
24         try{
25             Class.forName(JDBC_DRIVER);
26             Connection conn = DriverManager.getConnection(DB_URL, USER,
27                 PASS);
28
29             String deleteCmd = "DELETE FROM Records WHERE pid = \"" + pid
30                 + "\"";
31             Statement stmt = conn.createStatement();
32             stmt.execute(deleteCmd); //execute the Statement
33
34             //To open the view.html page after adding the record
35             RequestDispatcher rd = request.getRequestDispatcher("/view.
36                 html");
37             rd.forward(request, response);
38
39             stmt.close();
40             conn.close();
41         }
42
43         catch(SQLException se){
44             se.printStackTrace();
45             out.println(se);
46         }
47     }
48 }
```

```
43         catch(Exception e){
44             e.printStackTrace();
45             out.println(e);
46         }
47     }
48 }
```



## Code - Search Record Servlet:

```
1 import java.io.*;
2 import java.sql.*;
3 import java.util.*;
4 import javax.servlet.*;
5 import javax.servlet.http.*;
6
7 public class SearchRecord extends HttpServlet{
8     public void doGet(HttpServletRequest request, HttpServletResponse
9         response)
10         throws ServletException, IOException{
11
12         //DB_URL: jdbc:mysql:location:port/DatabaseName?Parameters
13
14         String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
15         String DB_URL = "jdbc:mysql://localhost:3306/Patient_Details?
16         autoReconnect=true&useSSL=false";
17         String USER = "admin";
18         String PASS = "password";
19
20         PrintWriter out = response.getWriter();
21         response.setContentType("text/html");
22
23         String pid = request.getParameter("pid");
24
25         try{
26             Class.forName(JDBC_DRIVER);
27             Connection conn = DriverManager.getConnection(DB_URL, USER,
28                 PASS);
29
30             String searchCmd = "SELECT * FROM Records WHERE pid = \"" +
31                 pid + "\"";
32             Statement stmt = conn.createStatement();
33
34             ResultSet rs = stmt.executeQuery(searchCmd);
35
36             if(rs.next()){
37                 pid = rs.getString("pid");
38                 String pname = rs.getString("pname");
39                 int age = rs.getInt("age");
40                 String gender = rs.getString("gender");
41
42                 switch(gender){
43                     case "M":
44                         gender = "Male";
45                         break;
46                     case "F":
47                         gender = "Female";
```

```

44         break;
45     case "0":
46         gender = "Other";
47         break;
48     default:
49         gender = "Null";
50 }
51
52 String mstatus = rs.getString("mstatus");
53 String addr = rs.getString("addr");
54 long phone = rs.getLong("phone");
55 String dov = rs.getString("dov");
56
57 out.println("<!DOCTYPE HTML>");
58 out.println("<html><head><title>Patient Details</title><
link rel='stylesheet' href='css/pages.css'></head>");
59 out.println("<body><h1 class='titleText'>Patient Details</
h1><hr><br><table align='center'>");
60 out.println("<tr><th>Patient ID</th><th>Name</th><th>Age</
th><th>Gender</th>" +
61 "<th>Marital Status</th><th>Address</th><th>Phone</th><th>
Date of Visit</th></tr>");
62
63 out.println("<tr><td>" + pid + "</td><td>" + pname + "</td
><td>" + age + "</td><td>" + gender +
64 "</td><td>" + mstatus + "</td><td>" + addr + "</td><td>" +
phone + "</td><td>" + dov + "</td></tr>");
65
66 out.println("</body></html>");
67
68 }
69
70 else{
71     out.println("<!DOCTYPE HTML>");
72     out.println("<html><head><link rel='stylesheet' href='css/
pages.css'></head>");
73     out.println("<body><h1 class='titleText'>Record with the
ID: " + pid + " does not exist.</h1><br>");
74     out.println("</body></html>");
75 }
76
77 stmt.close();
78 conn.close();
79 }
80
81 catch(SQLException se){
82     se.printStackTrace();
83     out.println(se);
84 }
85
86 catch(Exception e){

```

```
87         e.printStackTrace();
88         out.println(e);
89     }
90 }
91 }
```

## Code - View Records Servlet:

```
1 import java.io.*;
2 import java.sql.*;
3 import java.util.*;
4 import javax.servlet.*;
5 import javax.servlet.http.*;
6
7 public class ViewRecords extends HttpServlet{
8     public void doGet(HttpServletRequest request, HttpServletResponse
9         response)
10         throws ServletException, IOException{
11         //DB_URL: jdbc:mysql:location:port/DatabaseName?Parameters
12
13         String JDBC_DRIVER = "com.mysql.cj.jdbc.Driver";
14         String DB_URL = "jdbc:mysql://localhost:3306/Patient_Details?
15         autoReconnect=true&useSSL=false";
16         String USER = "admin";
17         String PASS = "password";
18
19         PrintWriter out = response.getWriter();
20         response.setContentType("text/html");
21
22         try{
23             Class.forName(JDBC_DRIVER);
24             Connection conn = DriverManager.getConnection(DB_URL, USER,
25                 PASS);
26
27             String viewCmd = "SELECT * FROM Records";
28             //to navigate forwards and backwards, and also to make the
29             result set read-only
30             Statement stmt = conn.createStatement(ResultSet.
31                 TYPE_SCROLL_SENSITIVE, ResultSet.CONCUR_READ_ONLY);
32
33             ResultSet rs = stmt.executeQuery(viewCmd);
34
35             if(rs.first()){ //to check if the database has any rows
36
37                 rs.beforeFirst(); //to set the cursor back before the
38                 first row
39
40                 out.println("<!DOCTYPE HTML>");
41                 out.println("<html><head><title>Patient Details</title><
42                 link rel='stylesheet' href='css/pages.css'></head>");
43                 out.println("<body><h1 class='titleText'>Patient Details</
44                 h1><hr><br><table align='center'>");
45                 out.println("<tr><th>Patient ID</th><th>Name</th><th>Age</
46                 th><th>Gender</th>" +
```

```

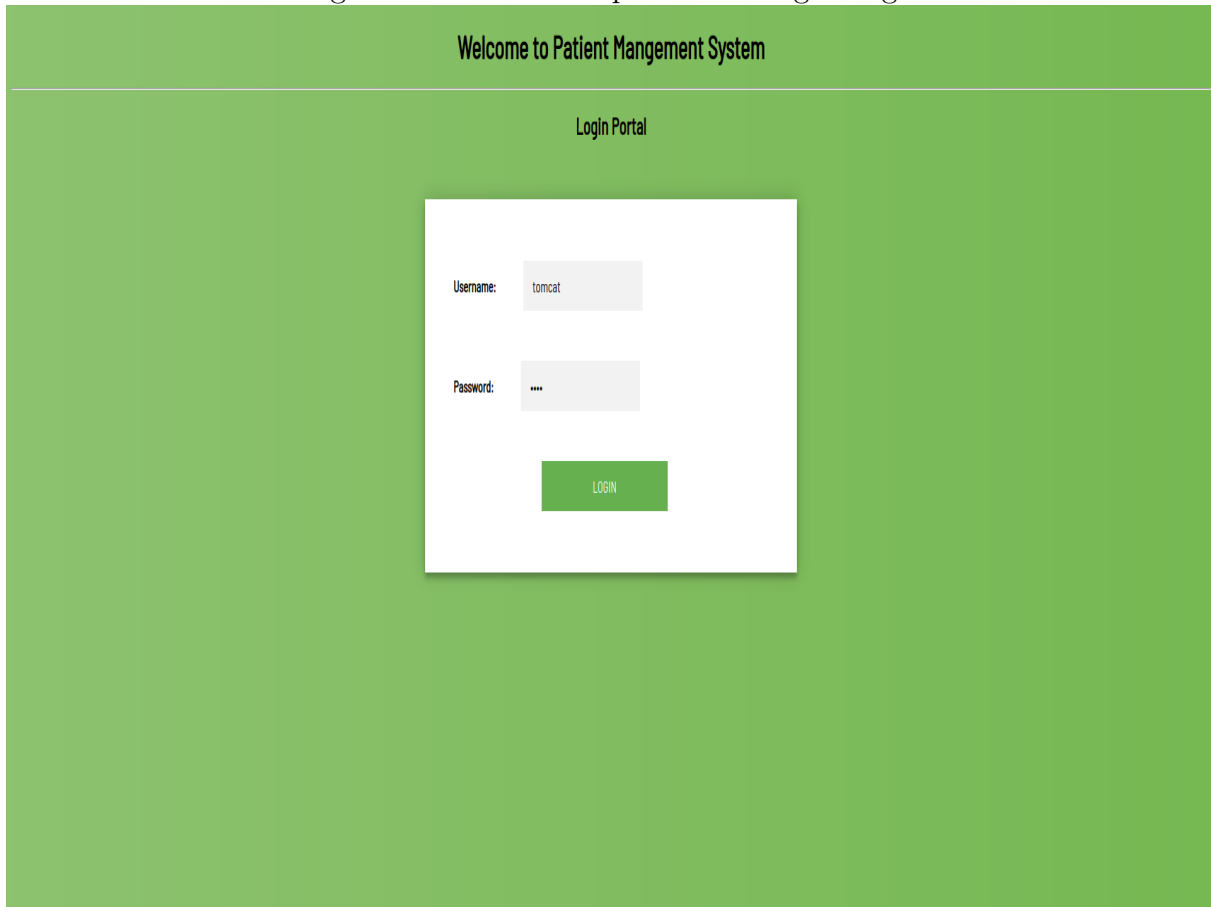
39         "<th>Marital Status</th><th>Address</th><th>Phone</th><th>
Date of Visit</th></tr>");
40
41         while(rs.next()){
42             String pid = rs.getString("pid");
43             String pname = rs.getString("pname");
44             int age = rs.getInt("age");
45             String gender = rs.getString("gender");
46
47             switch(gender){
48                 case "M":
49                     gender = "Male";
50                     break;
51                 case "F":
52                     gender = "Female";
53                     break;
54                 case "O":
55                     gender = "Other";
56                     break;
57                 default:
58                     gender = "Null";
59             }
60
61             String mstatus = rs.getString("mstatus");
62             String addr = rs.getString("addr");
63             long phone = rs.getLong("phone");
64             String dov = rs.getString("dov");
65
66             out.println("<tr><td>" + pid + "</td><td>" + pname + "
</td><td>" + age + "</td><td>" + gender +
67                 "</td><td>" + mstatus + "</td><td>" + addr + "</td><td>
>" + phone + "</td><td>" + dov + "</td></tr>");
68
69             }
70
71             out.println("</table></body></html>");
72
73         }
74
75         else{
76             out.println("The database is empty.<br>");
77         }
78
79
80         stmt.close();
81         conn.close();
82     }
83
84     catch(SQLException se){
85         se.printStackTrace();
86         out.println(se);

```

```
87         }
88
89         catch (Exception e){
90             e.printStackTrace();
91             out.println(e);
92         }
93     }
94 }
```

## Output - PMS Login Page:

Figure 3: Browser Output: PMS Login Page.



The screenshot displays a web browser window with a solid green background. At the top, a dark green header bar contains the text "Welcome to Patient Mangement System" in white. Below this, the text "Login Portal" is centered. A white login form is centered on the page, featuring a "Username:" label next to a text input field containing "tomcat", a "Password:" label next to a password input field with four dots, and a green "LOGIN" button at the bottom.

Welcome to Patient Mangement System

Login Portal

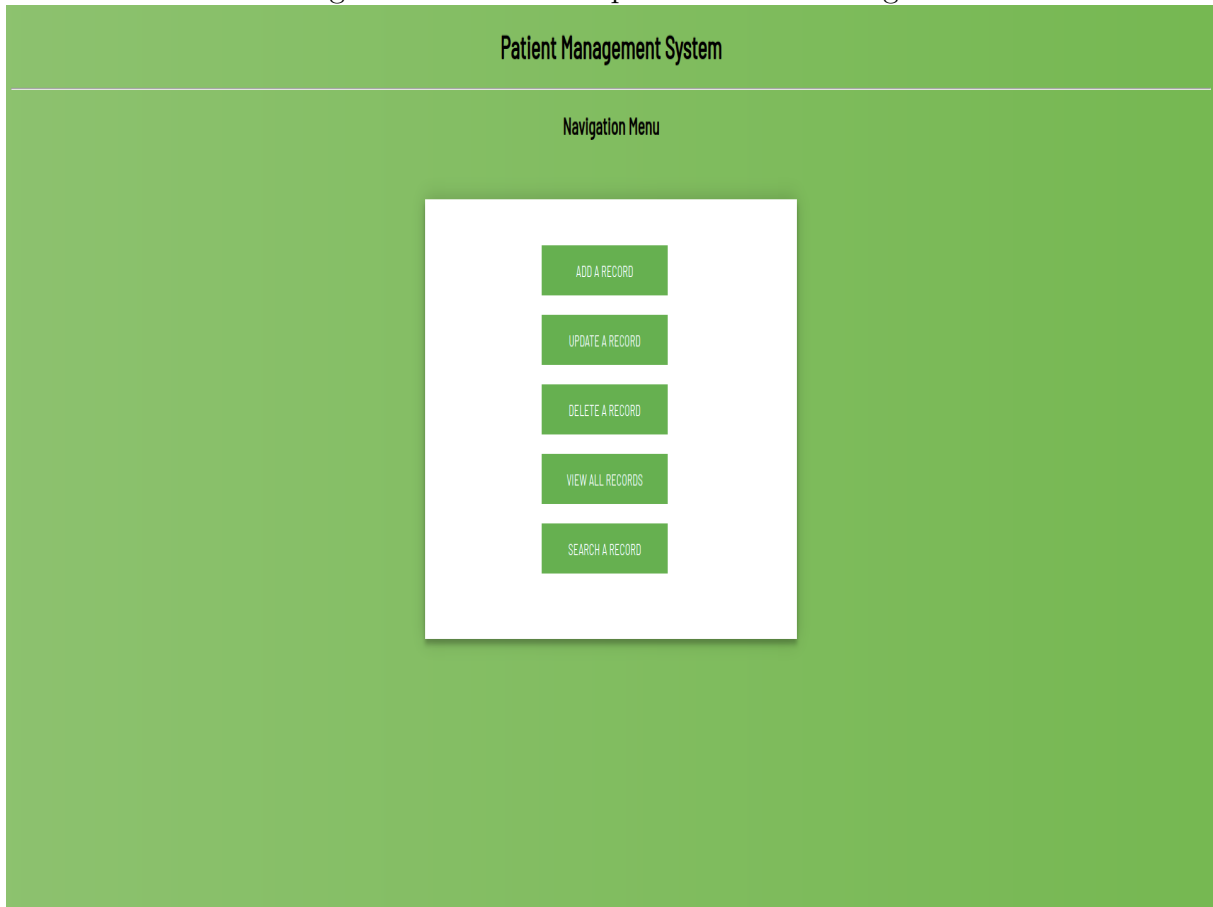
Username: tomcat

Password: \*\*\*\*

LOGIN

## Output - PMS Home Page:

Figure 4: Browser Output: PMS Home Page.





## Output - PMS Add Record Page:

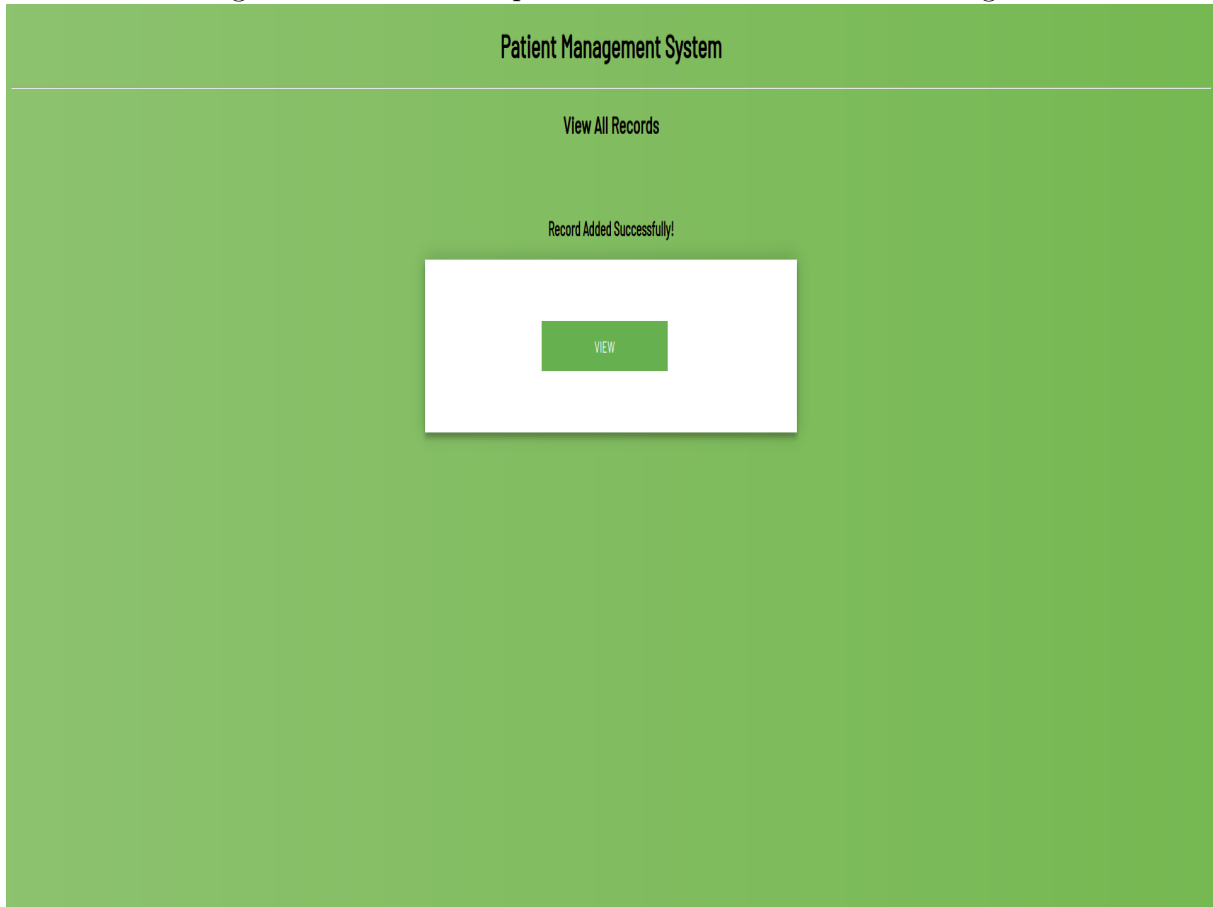
Figure 5: Browser Output: PMS Add Record Page.

The screenshot displays a web application titled "Patient Management System" with a sub-header "Add A Record". The form contains the following fields and controls:

- Patient ID:** Text input field containing "P2006".
- Name:** Text input field containing "Michelle Obama".
- Age:** Text input field containing "46".
- Gender:** Radio button group with options: ☐ Male, ☒ Female, ☐ Other.
- Marital Status:** Dropdown menu showing "Married".
- Address:** Text input field containing "1600, Pennsylvania Ave NW, Washington DC".
- Phone Number:** Text input field containing "8887654321".
- Date of Visit:** Text input field containing "17/01/1964" with a calendar icon.
- Buttons:** Two green buttons at the bottom: "REGISTER" and "CLEAR ALL".

## Output - PMS Add Success Page:

Figure 6: Browser Output: PMS Add Record Success Page.



## Output - PMS View Page:

Figure 7: Browser Output: PMS View Page.

Patient Details							
Patient ID	Name	Age	Gender	Marital Status	Address	Phone	Date of Visit
P2001	Chris Paley	25	Male	Single	275, Sunset Blvd, Los Angeles, CA	9887521131	2021-01-21
P2002	Rick Riordan	45	Male	Married	221E, Carpenter St, London, UK	9876544321	2021-02-23
P2004	Catalina Grace	34	Female	In A Relationship	34, King St, Mountain View, CA	9765535461	2020-09-18
P2006	Michelle Obama	46	Female	Married	1600, Pennsylvania Ave NW, Washington DC	8887654321	1964-01-17

## Output - PMS Update Record Page:

Figure 8: Browser Output: PMS Update Record Page.

The screenshot displays a web application interface for a Patient Management System. The page has a green header with the title "Patient Management System". Below the header, the main content area is also green and contains the text "Update A Record". In the center of the page is a white form box. The form contains three input fields: "Patient ID" with the value "P2006", "Updated Address" with the value "2500W, W. Golf Road, Hoffman Estates, New Jersey", and "Updated Phone Number" with the value "8765944321". At the bottom of the form are two green buttons: "UPDATE" and "CLEAR ALL".

Patient Management System

Update A Record

Patient ID

P2006

Updated Address

2500W, W. Golf Road, Hoffman  
Estates, New Jersey

Updated Phone Number

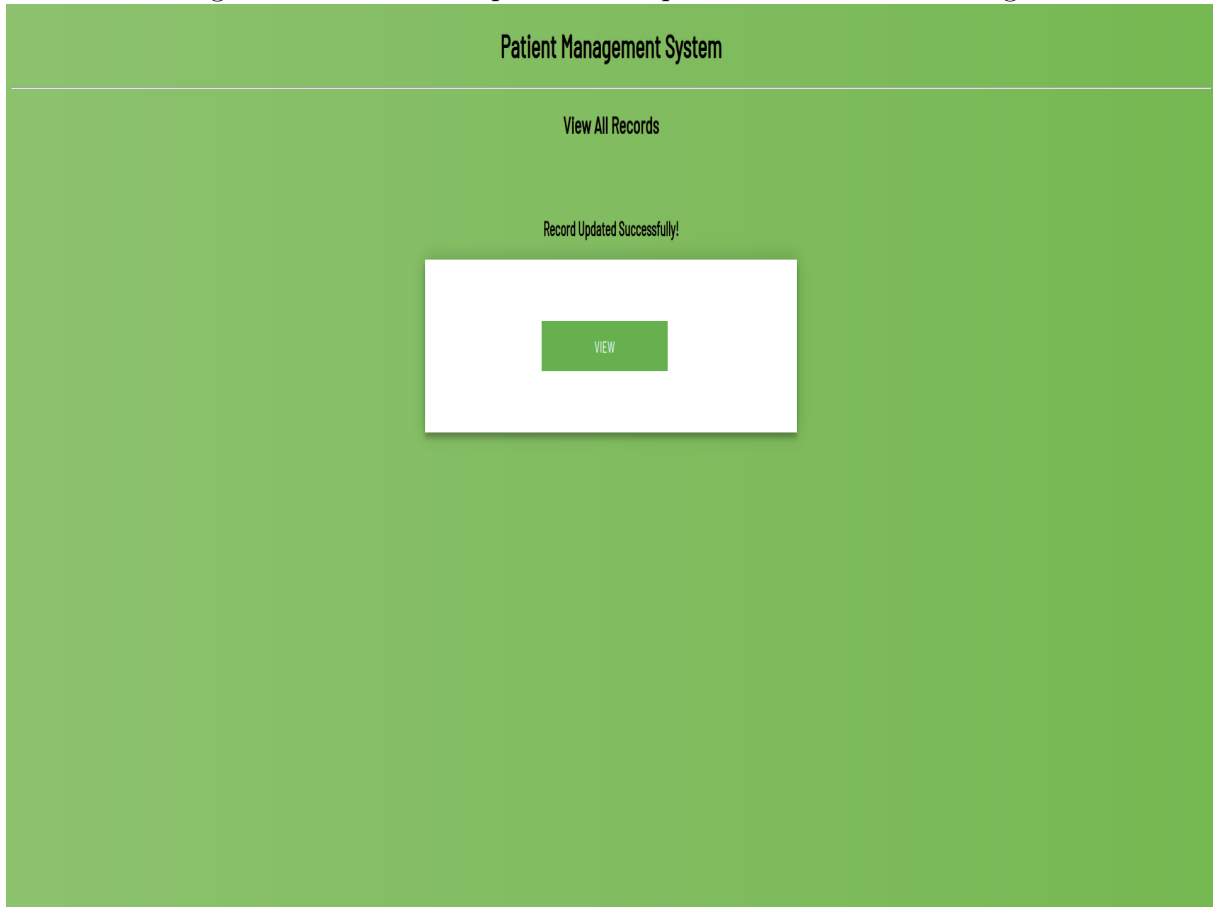
8765944321

UPDATE

CLEAR ALL

## Output - PMS Update Success Page:

Figure 9: Browser Output: PMS Update Record Success Page.



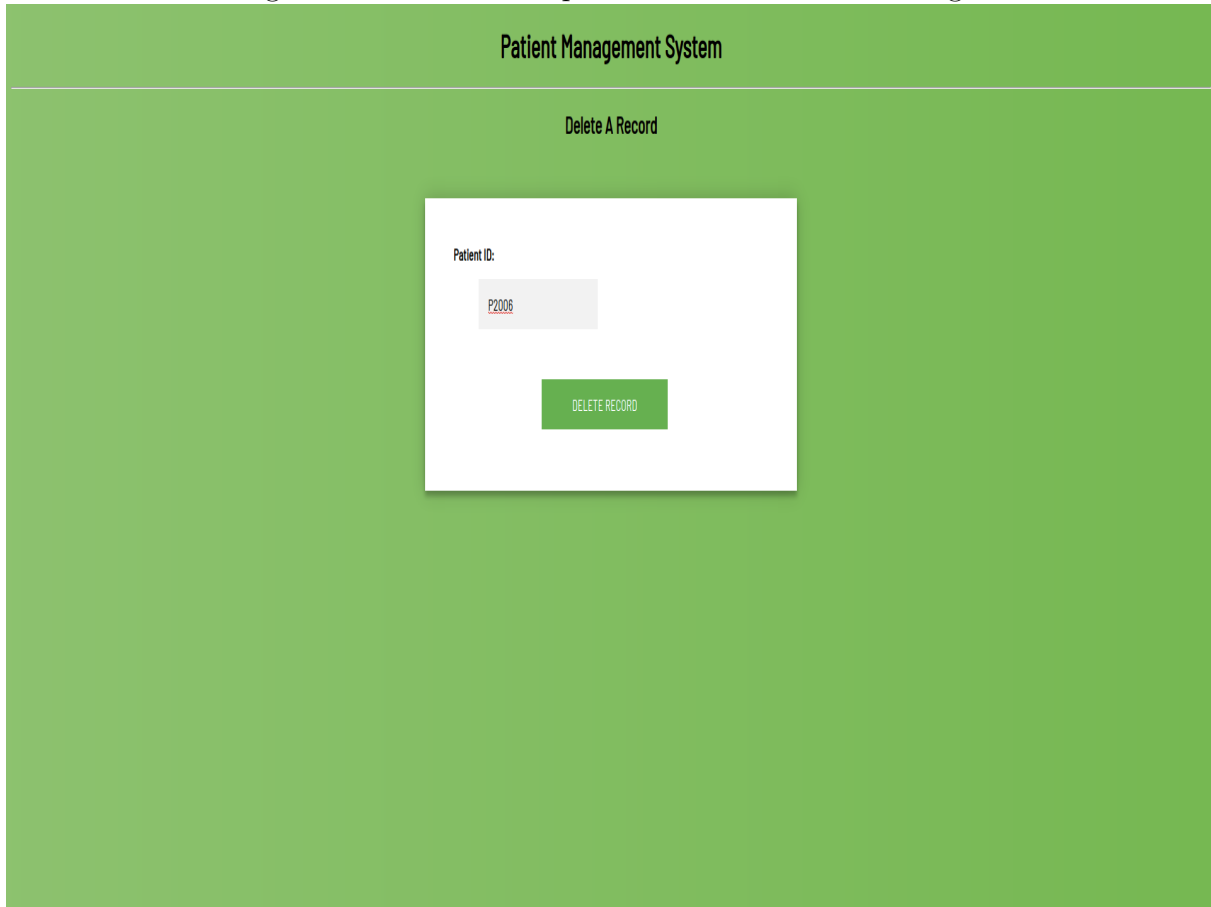
## Output - PMS View Page:

Figure 10: Browser Output: PMS View Page.

Patient Details							
Patient ID	Name	Age	Gender	Marital Status	Address	Phone	Date of Visit
P2001	Chris Paley	25	Male	Single	275, Sunset Blvd, Los Angeles, CA	9887521131	2021-01-21
P2002	Rick Riordan	45	Male	Married	221E, Carpenter St, London, UK	9876544321	2021-02-23
P2004	Catalina Grace	34	Female	In A Relationship	34, King St, Mountain View, CA	9765535461	2020-09-18
P2006	Michelle Obama	46	Female	Married	2500W, W. Golf Road, Hoffman Estates, New Jersey	8765944321	1964-01-17

## Output - PMS Delete Record Page:

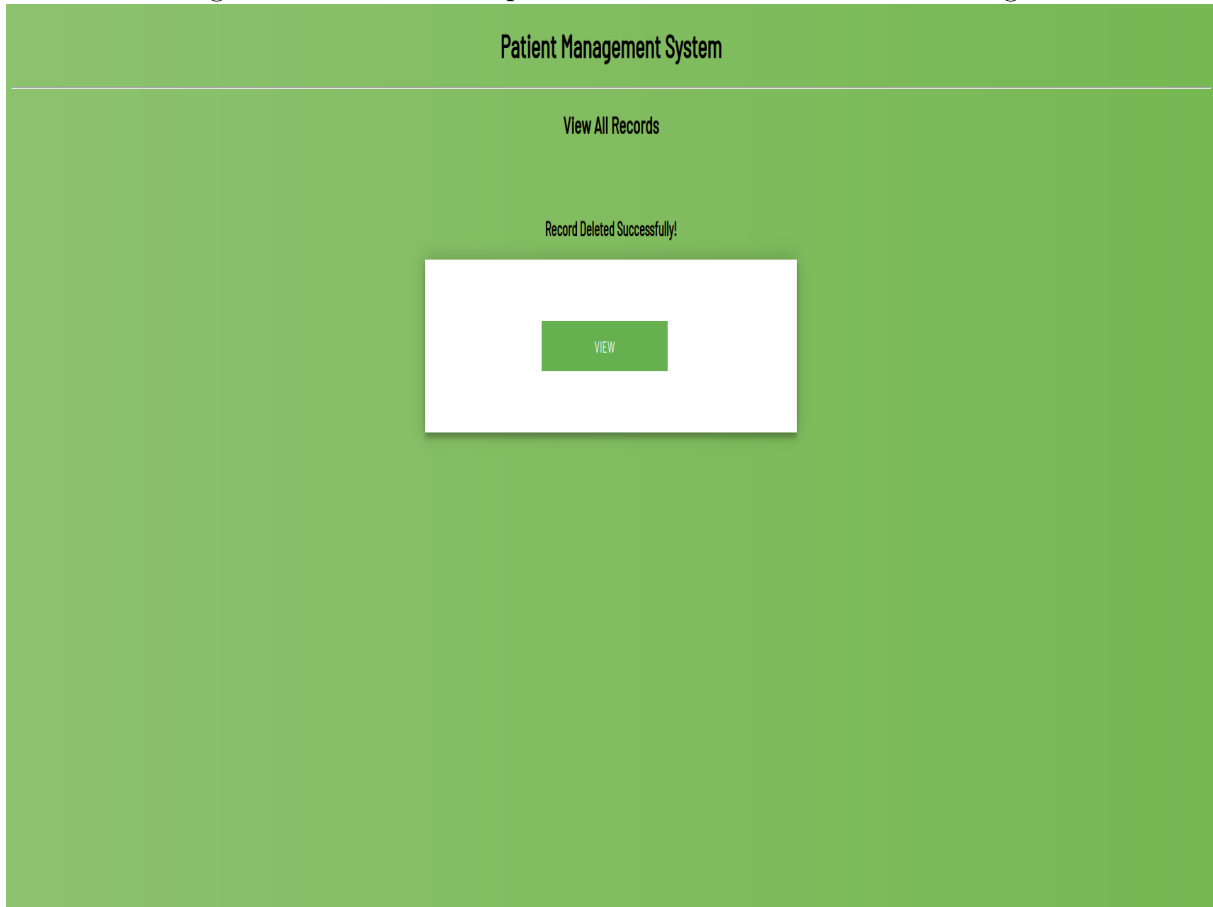
Figure 11: Browser Output: PMS Delete Record Page.



The screenshot displays a web application interface for a Patient Management System. The page has a solid green background. At the top, the text "Patient Management System" is centered in a white font. Below this, the text "Delete A Record" is centered in a white font. In the center of the page, there is a white rectangular form. Inside this form, the text "Patient ID:" is displayed. Below it, there is a text input field containing the value "P2008". At the bottom of the form, there is a green button with the text "DELETE RECORD" in white capital letters.

## Output - PMS Delete Success Page:

Figure 12: Browser Output: PMS Delete Record Success Page.





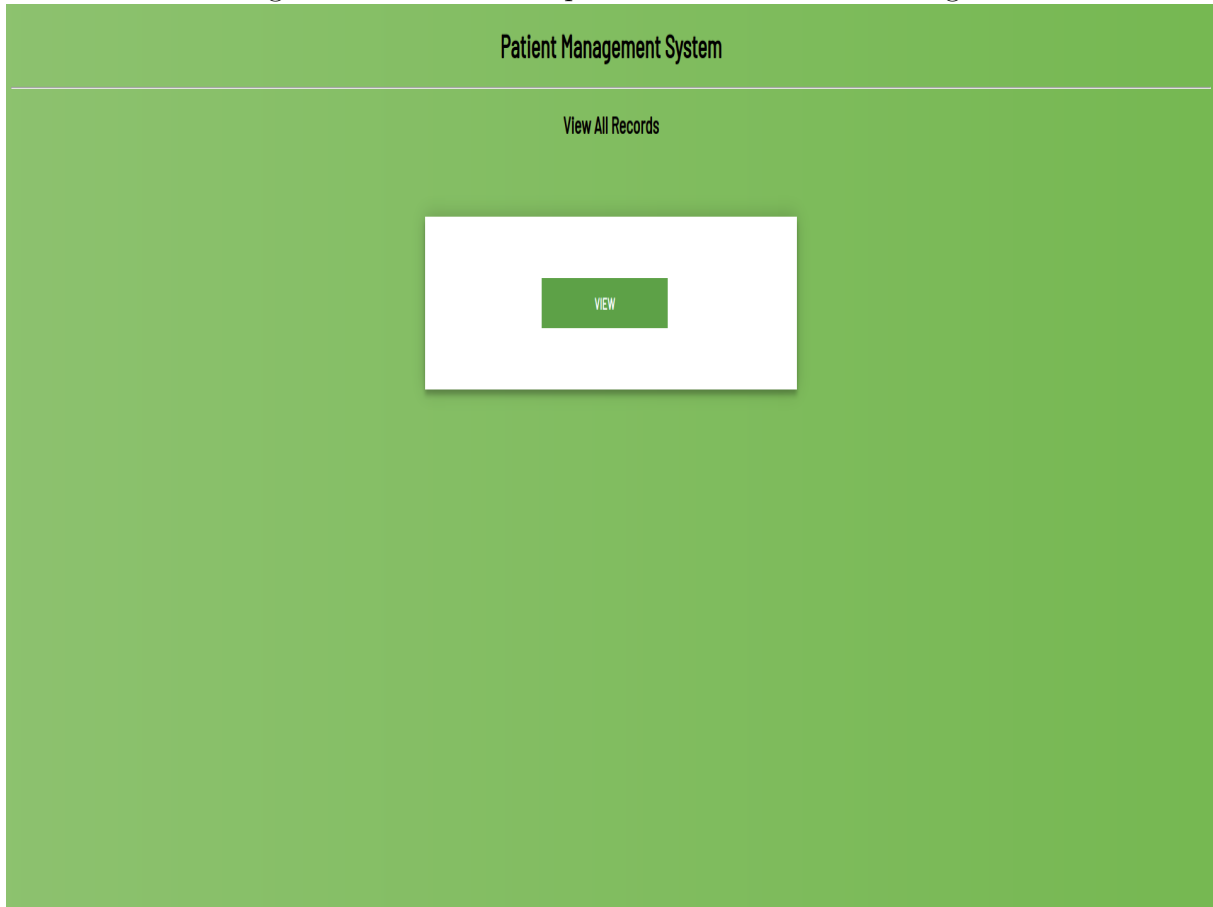
## Output - PMS View Page:

Figure 13: Browser Output: PMS View Page.

Patient Details							
Patient ID	Name	Age	Gender	Marital Status	Address	Phone	Date of Visit
P2001	Chris Paley	25	Male	Single	275, Sunset Blvd, Los Angeles, CA	9887521131	2021-01-21
P2002	Rick Riordan	45	Male	Married	221E, Carpenter St, London, UK	9876544321	2021-02-23
P2004	Catalina Grace	34	Female	In A Relationship	34, King St, Mountain View, CA	9765535461	2020-09-18
P2006	Michelle Obama	46	Female	Married	2500W, W. Golf Road, Hoffman Estates, New Jersey	8765944321	1964-01-17

## Output - PMS View Records Page:

Figure 14: Browser Output: PMS View Records Page.



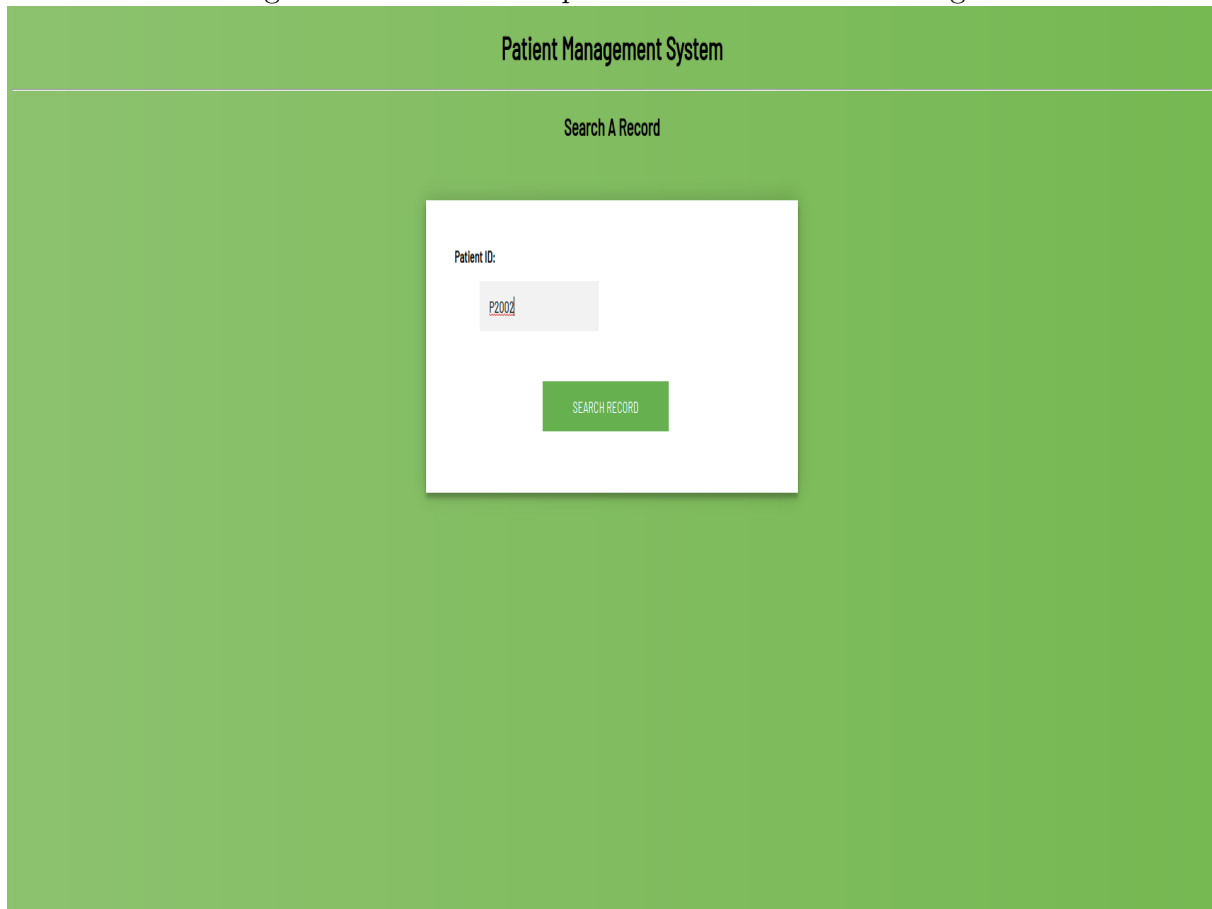
## Output - PMS View Table Page:

Figure 15: Browser Output: PMS View Table Page.

Patient Details							
Patient ID	Name	Age	Gender	Marital Status	Address	Phone	Date of Visit
P2001	Chris Paley	25	Male	Single	275, Sunset Blvd, Los Angeles, CA	9887521131	2021-01-21
P2002	Rick Riordan	45	Male	Married	221E, Carpenter St, London, UK	9876544321	2021-02-23
P2004	Catalina Grace	34	Female	In A Relationship	34, King St, Mountain View, CA	9765535461	2020-09-18

## Output - PMS Search Record Page:

Figure 16: Browser Output: PMS Search Record Page.



The screenshot displays a web application interface for a Patient Management System. The page has a solid green background. At the top, the text "Patient Management System" is centered in a white, sans-serif font. Below this, the text "Search A Record" is centered in a smaller, white, sans-serif font. In the center of the page, there is a white rectangular form with a thin black border. Inside this form, the text "Patient ID:" is positioned above a text input field. The input field contains the text "P2000" in a red, monospaced font. Below the input field, there is a green rectangular button with the text "SEARCH RECORD" in a white, sans-serif font.

## Output - PMS Search Success Page:

Figure 17: Browser Output: PMS Search Record Success Page.

Patient Details							
Patient ID	Name	Age	Gender	Marital Status	Address	Phone	Date of Visit
P2002	Rick Riordan	45	Male	Married	221E, Carpenter St, London, UK	9876544321	2021-02-23

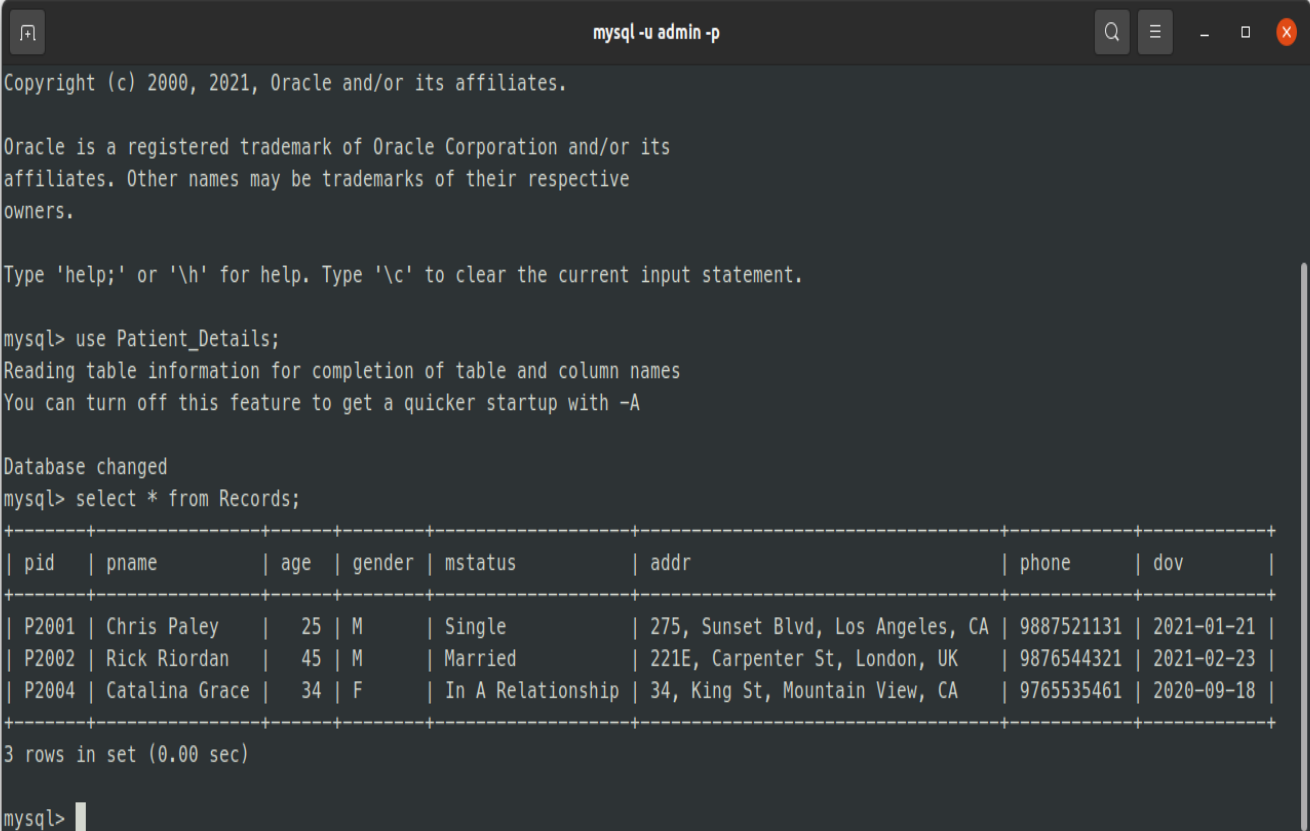
## Output - PMS View Page:

Figure 18: Browser Output: PMS View Page.

Patient Details							
Patient ID	Name	Age	Gender	Marital Status	Address	Phone	Date of Visit
P2001	Chris Paley	25	Male	Single	275, Sunset Blvd, Los Angeles, CA	9887521131	2021-01-21
P2002	Rick Riordan	45	Male	Married	221E, Carpenter St, London, UK	9876544321	2021-02-23
P2004	Catalina Grace	34	Female	In A Relationship	34, King St, Mountain View, CA	9765535461	2020-09-18

## Output - SQL Table:

Figure 19: Terminal Output: SQL Table.



```
mysql -u admin -p
Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use Patient_Details;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from Records;
```

pid	pname	age	gender	mstatus	addr	phone	dov
P2001	Chris Paley	25	M	Single	275, Sunset Blvd, Los Angeles, CA	9887521131	2021-01-21
P2002	Rick Riordan	45	M	Married	221E, Carpenter St, London, UK	9876544321	2021-02-23
P2004	Catalina Grace	34	F	In A Relationship	34, King St, Mountain View, CA	9765535461	2020-09-18

```
3 rows in set (0.00 sec)

mysql>
```

## Learning Outcome:

- From the experiment, I learnt to implement basic **Java Servlets**.
- I learnt about the working of a Java Servlet.
- I understood that Servlets are executed in the server-side in response to an HTML event like a submit event, and handles actions through a Request-Response mechanism.
- I was able to serve a dynamic web page using Java Servlet programming.
- I learnt how to extract HTML Form data from Servlets submitted using the GET query.
- I calculated the employee's gross pay from the basic salary entered in the form in server-side using Java Servlet and displayed it back to the client-side through the browser using HTML.
- I successfully implemented a working **Employee Registration Form** using Java Servlet programming and deployed it using the **Tomcat 9.0 Container**.
- I refreshed my SQL concepts.
- I learnt how to integrate MySQL to Java using JDBC Driver.
- I was able to create a database for Patient Management using MySQL.
- I successfully implemented JDBC connectivity to MySQL.
- I implemented the management operations like Add, Delete, Update and Viewing Records from the client-side, and updated the back-end SQL database using Java Servlets, with the help of JDBC API calls.
- I learnt to authorize a user with basic UserID & Password using JavaScript.
- I learnt to serve an existing HTML page from a Java Servlet.
- I understood how to handle basic SQLExceptions with the help of the try-catch block in Java.
- I understood how to display records dynamically with the help of ResultSet Object in Java and <table> element in HTML.
- I was able to successfully implement a working **Patient Management System** according to the given specifications with Java Servlets and MySQL.