**1. INTRODUCTION**

[Mobiles shopping](http://en.wikipedia.org/wiki/Online_shopping) is a form of electronic commerce where the buyer is directly online to the seller's computer usually via the internet. There is no intermediary service. The sale and purchase transaction is completed electronically and interactively in real-time such as Amazon.com.

The central concept of the application is to allow the customer to shop virtually using the Internet and allow customers to buy the Mobiles Phones of their desire and recharge their internet account from the store. The information pertaining to the products are stores on an RDBMS at the server side (store). The Server process the customers and the items are shipped to the address submitted by them. The application was designed into three modules first is for the customers who wish to buy the articles. Second is for the admin who maintains and updates the information pertaining to the articles and those of the customers.

**2. SCOPE**

Still in shops all data processing is done manually. When there are a lot of issues such as retrieval and storage of the information and keeping track of them becomes a tedious task. By implementing a computerized system, the limitation in the present system will be reduced. Manpower can be reduced to a great extent and efficiency and accuracy can be increased to manifold. More over consumption of time can be reduced to far greater extend.

This project aims at eliminating the limitations of the increasing speed of information retrieval, by increasing processing speed, data security and hence give accurate reports. The proposed system fully satisfies user needs. All most all the limitations of the existing system are reduced in a great manner. Details are stored in various files from which data can be accessed and manipulated easily.

The project is used to replace the manual system with the computerized system. It will help to save a lot of time, money and energy.

**3. REQUIREMENTS**

**Hardware Requirement**

Processor : Intel Core Duo 2.0 GHz or more

RAM : 1 GB or More

Hard disk : 80GB or more

Monitor : 15” CRT or LCD monitor

Keyboard : Normal or Multimedia

Mouse : Compatible mouse

**Software Requirement**

**Server-Side**

Front End : Java

Back End : MySql 5.6

Operating System : Windows 7/8 Professional

Workbench : Eclipse Kepler

Server : Apache TomCat 7

**Client-Side**

Front End : Java , JavaScript, HTML

Back End : MySql 5.6

Operating System : Windows 7/8 Professional

Workbench : Eclipse Kepler

**4. ER DIAGRAM AND DATA FLOW DIAGRAM**

**4.1. E-R DIAGRAM**

feedback

Admin\_ Login

give

Data\_plans

Hello.com

Customer

mobile\_details

Will buy

Fig4.1: E-R Diagram

**4.2. DATA FLOW DIAGRAM**

**Delete,View**

**Update**

**ADMIN**

**feedback**

**Buy mobile**

**Users**

**Users**

**Recharge**

Fig4.2: DataFlow Diagram

**5. RELATIONAL DATABASE DESIGN**

**5.1 Table Creation Queries:**

CREATE TABLE customer ( `name` VARCHAR(45) NOT NULL ,`address` VARCHAR(60) NOT NULL ,`ph` VARCHAR(45) NOT NULL , `email` VARCHAR(45) NOT NULL ,`city` VARCHAR(45) NOT NULL ,`date` VARCHAR(45) NOT NULL ,`mobile` VARCHAR(45) NOT NULL , `price` VARCHAR(45) NOT NULL );

CREATE TABLE data\_plans (`plan\_no` varchar(12) NOT NULL,`plan\_name` varchar(45) DEFAULT NULL,`validity` varchar(45) DEFAULT NULL,`data\_usage` varchar(45) DEFAULT NULL,`speed` varchar(45) DEFAULT NULL,`charges` varchar(45) DEFAULT NULL,PRIMARY KEY (`plan\_no`));

CREATE TABLE internet\_order (`first\_name` varchar(45) DEFAULT NULL,`last\_name` varchar(45) DEFAULT NULL, `street` varchar(45) DEFAULT NULL, `city` varchar(45) DEFAULT NULL,`state` varchar(45) DEFAULT NULL, `zip\_code` varchar(31) DEFAULT NULL,`phone` varchar(30) DEFAULT NULL,`credit\_no` varchar(50) DEFAULT NULL,`expiry` varchar(45) DEFAULT NULL, `expiry2` varchar(45) DEFAULT NULL );

CREATE TABLE new\_subscribe (`form\_id` varchar(10) NOT NULL,`name` varchar(45) DEFAULT NULL, `address` varchar(45) DEFAULT NULL,`email` varchar(45) DEFAULT NULL,`phone` varchar(45) DEFAULT NULL,`dob` varchar(45) DEFAULT NULL,`gender` varchar(45) DEFAULT NULL,`nationality` varchar(45) DEFAULT NULL,`proofs` varchar(45) DEFAULT NULL,PRIMARY KEY (`form\_id`));

CREATE TABLE login\_internet (`user\_name` varchar(30) NOT NULL,`password` varchar(45) NOT NULL,PRIMARY KEY (`user\_name`));

CREATE TABLE feedback3 (`feedback` VARCHAR(45) NOT NULL ,`feedback2` VARCHAR(45) NOT NULL ,`name` VARCHAR(45) NULL ,

`address` VARCHAR(45) NULL , `email` VARCHAR(45) NULL , `phone` VARCHAR(45) NULL ,`comments` VARCHAR(200) NULL );

CREATE TABLE detail (`id` INT(10) UNSIGNED NOT NULL ,`pn` VARCHAR(45) NOT NULL ,`cam` VARCHAR(45) NOT NULL );

CREATE TABLE feedback (`name` varchar(45) NOT NULL,`feed` varchar(150) NOT NULL) ;

CREATE TABLE main (`id` INT(10) UNSIGNED NOT NULL ,`brand` VARCHAR(45) NOT NULL , `pname` VARCHAR(45) NOT NULL ,`os` VARCHAR(45) NOT NULL ,`camera` VARCHAR(45) NOT NULL ,`stock` VARCHAR(45) NOT NULL ,`cost` VARCHAR(45) NOT NULL ,`size` VARCHAR(45) NOT NULL ,`colour` VARCHAR(45) NOT NULL ,`processor` VARCHAR(45) NOT NULL , `memory` VARCHAR(45) NOT NULL ,`RAM` VARCHAR(45) NOT NULL ,PRIMARY KEY (`id`) );

CREATE TABLE temp (`id` VARCHAR(45) NOT NULL );

CREATE TABLE temp2 ( `name` VARCHAR(45) NOT NULL ,`cost` VARCHAR(45) NOT NULL );

**6. GRAPHICAL USER INTERFACE (SCREEN SHOTS OF IMPOTANT FORMS)**

**6.1. Login Form**

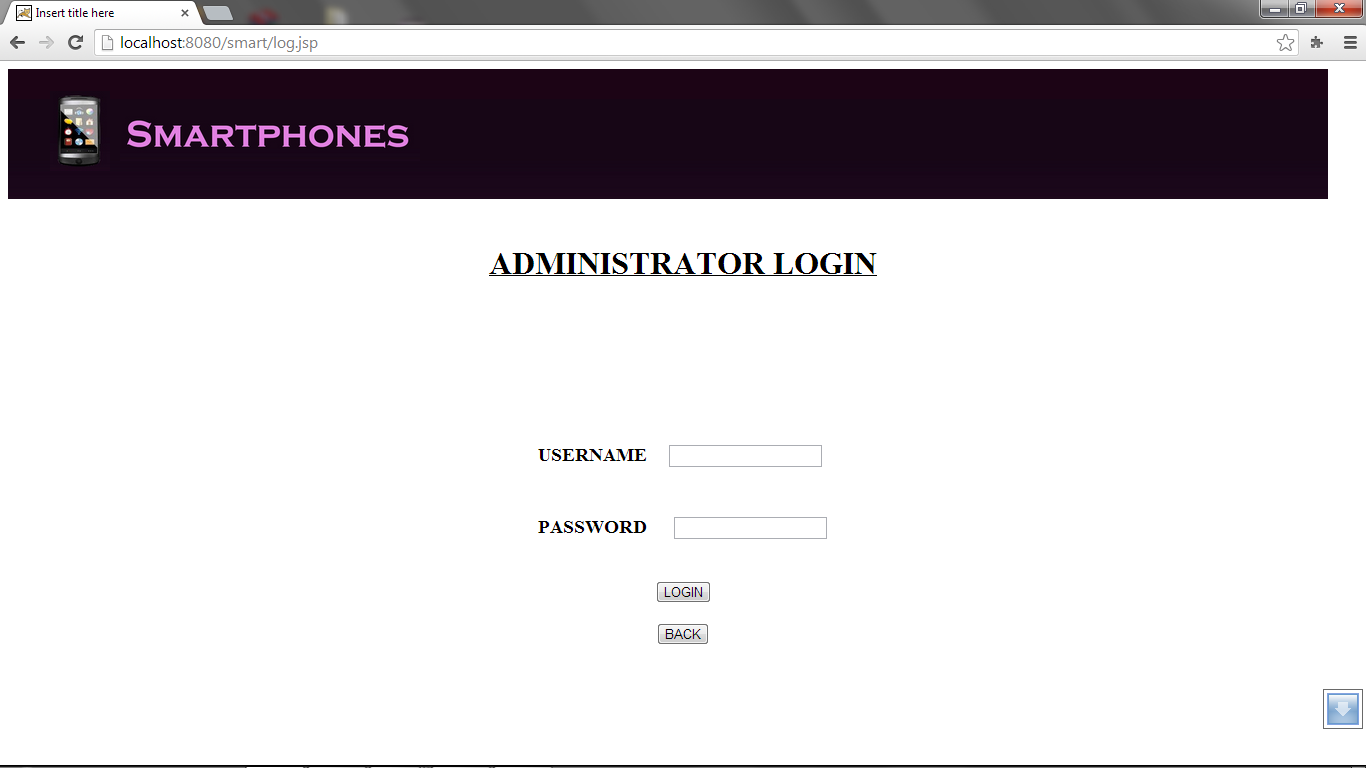


Fig6.1. Login Form

**6.2. Home Page**

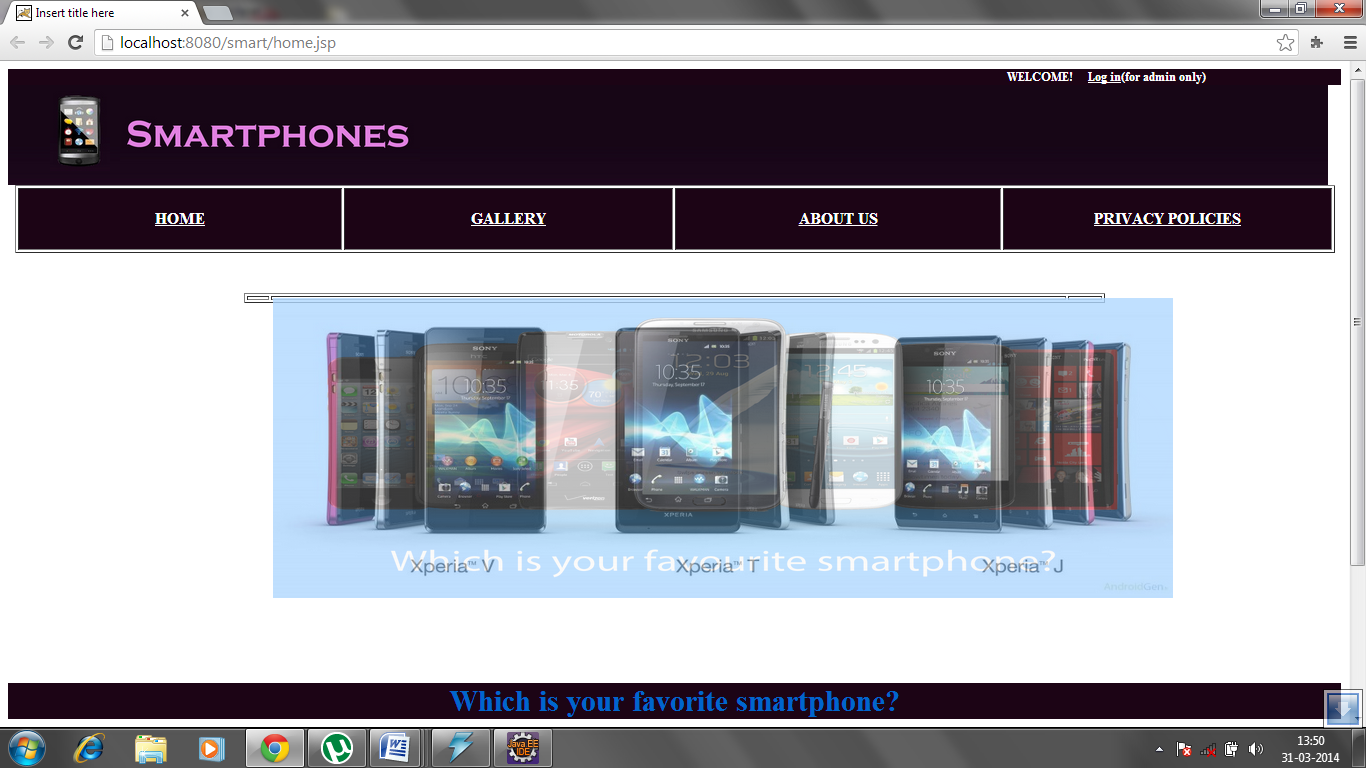
****

Fig6.2.Home Page

**6.3.Mobile brand page**

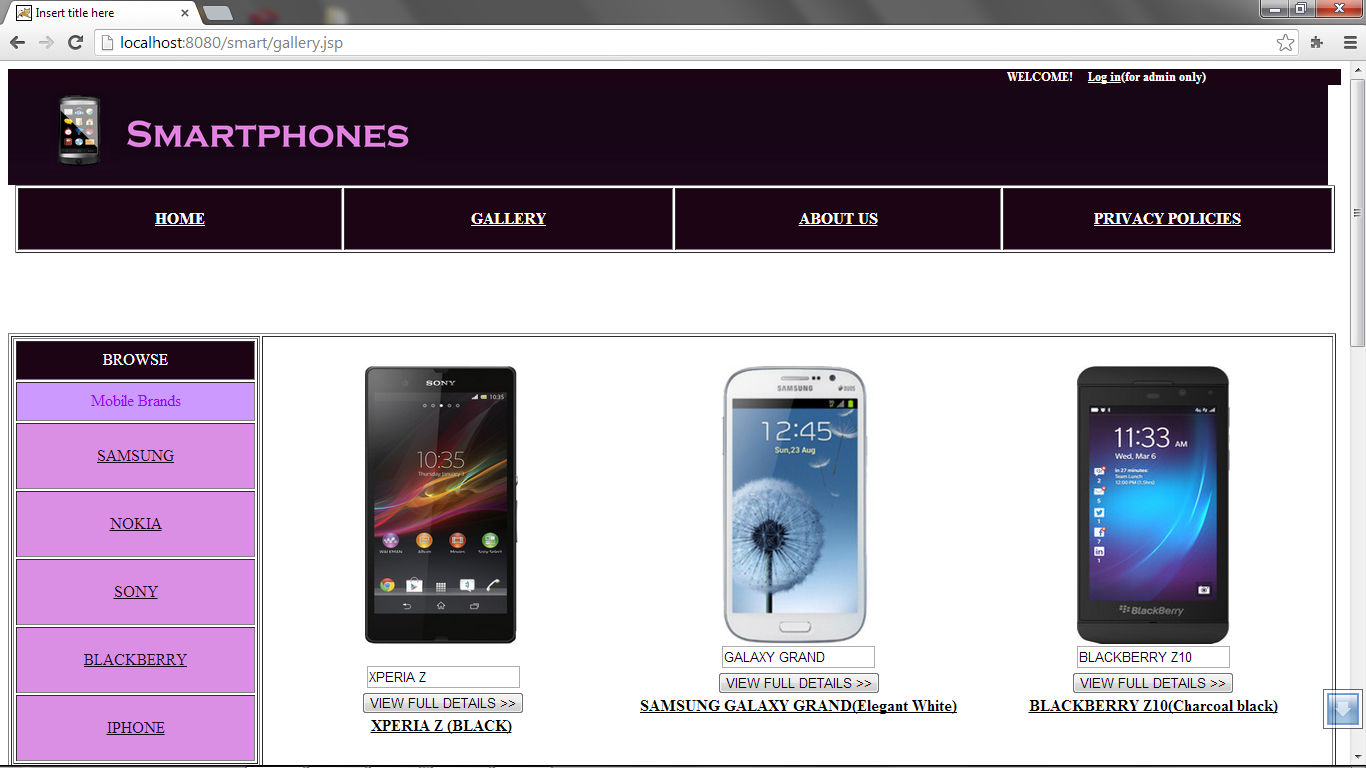
****

Fig6.3.Mobile brand page

**6.4. Mobile Order details Page**

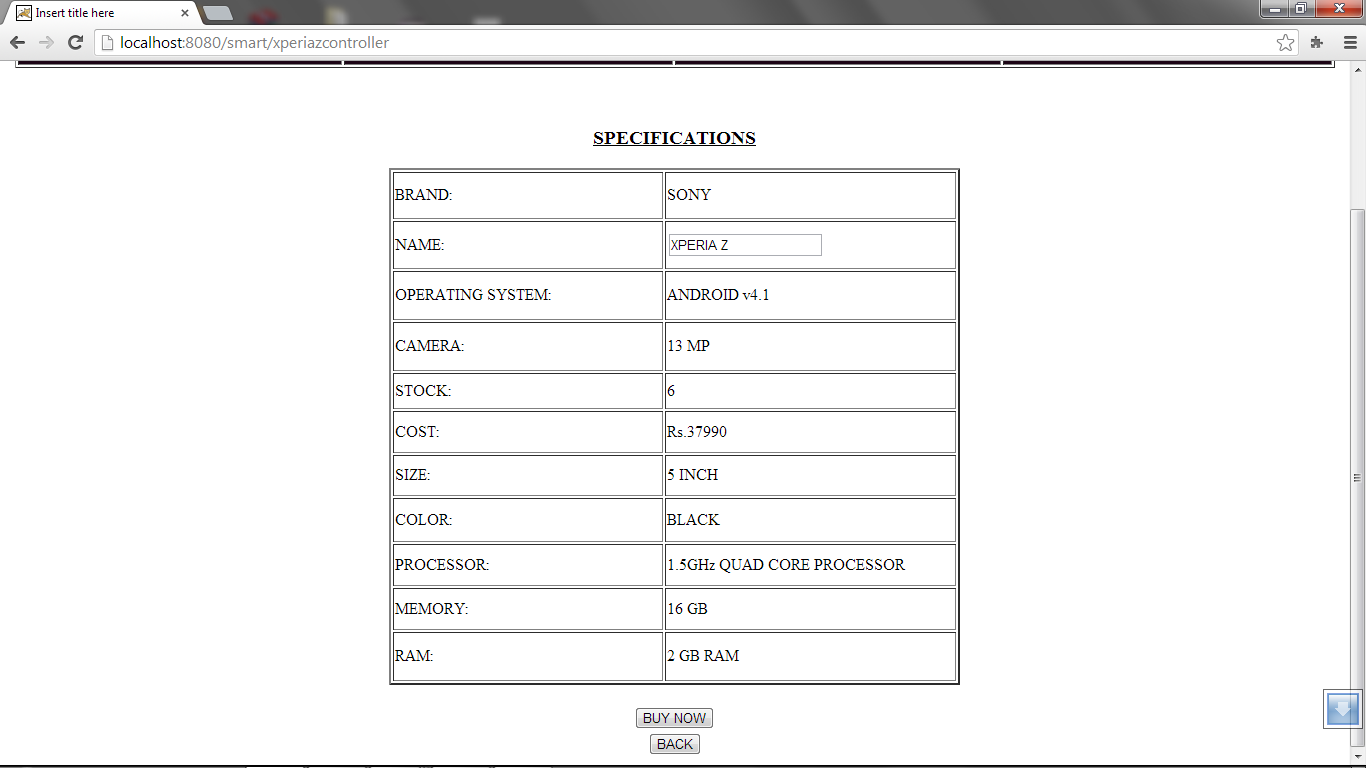
****

Fig6.4.Mobile Order details Page

**6.5. Order page**

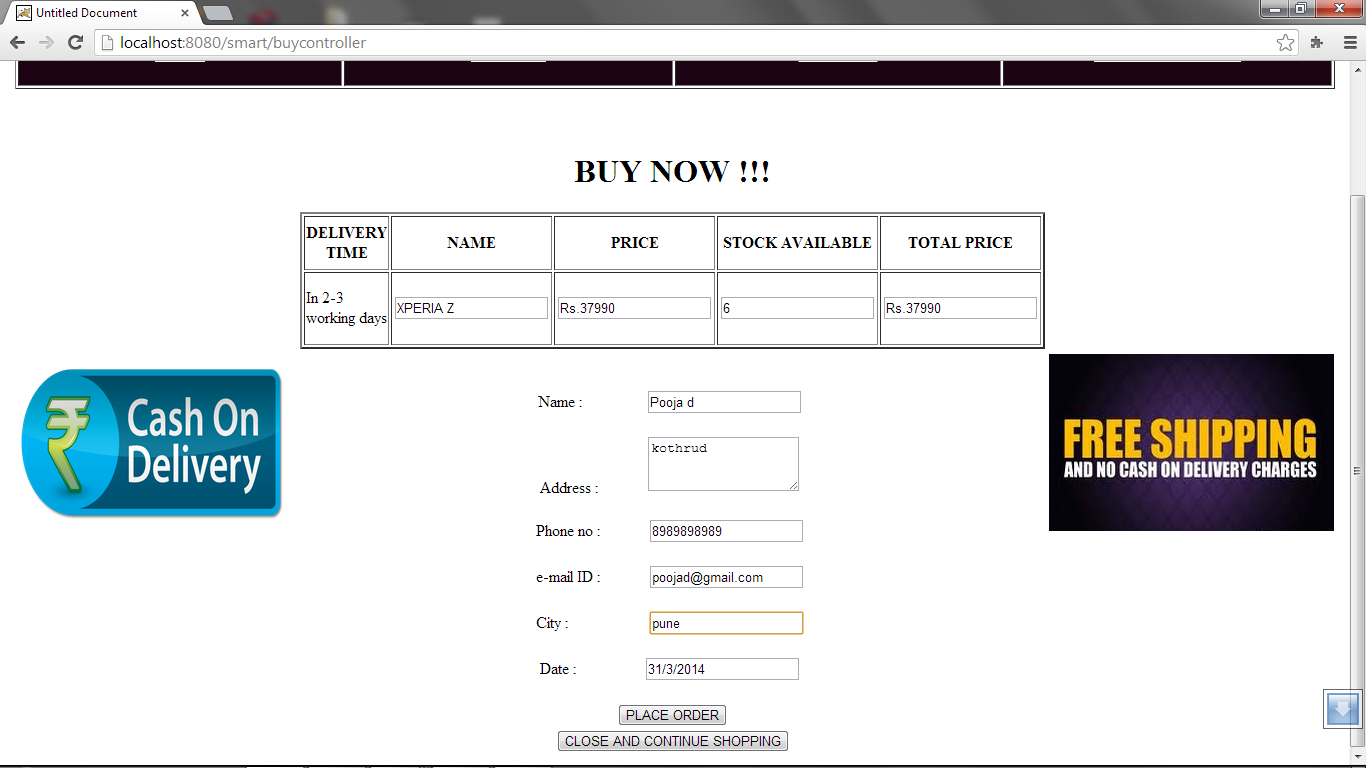
****

Fig6.5.Order page

1. **SOURCE CODE (CODE OF IMPORTANT FUNCTIONALITY)**

**7.1. For connectivity of forms**

**package** com.phone;

**import** java.io.IOException;

**import** javax.servlet.ServletException;

**import** javax.servlet.http.HttpServlet;

**import** javax.servlet.http.HttpServletRequest;

**import** javax.servlet.http.HttpServletResponse;

**import** java.sql.\*;

**import** java.io.\*;

**import** javax.servlet.\*;

**import** javax.servlet.http.\*;

**public** **class** logcontroller **extends** HttpServlet {

**private** **static** **final** **long** serialVersionUID = 1L;

/\*\*

\* **@see** HttpServlet#HttpServlet()

\*/

**public** logcontroller() {

**super**();

// **TODO** Auto-generated constructor stub

}

/\*\*

\* **@see** HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)

\*/

**protected** **void** doGet(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {

// **TODO** Auto-generated method stub

doPost (request,response);}

/\*\*

\* **@see** HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)

\*/

**protected** **void** doPost(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {

// **TODO** Auto-generated method stub

response.setContentType("text/html;charset=UTF-8");

PrintWriter out = response.getWriter();

**try**{

String name=request.getParameter("uname");

String pass=request.getParameter("password");

System.out.println("name="+name+"password="+pass);

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/login","root","ridhima");

Statement s=con.createStatement();

ResultSet rs=s.executeQuery("select \* from log where name='"+name+"'");

**if**(rs.next())

{

**if**( (name.trim().equals(rs.getString(1).trim())) && (

pass.trim().equals(rs.getString(2).trim()) ) )

{

//New session creation

HttpSession session=request.getSession(**true**);

//setting attribute on session

session.setAttribute("username",name);

//send request to Welcome.jsp page

RequestDispatcher view =

request.getRequestDispatcher("welcome.jsp");

view.forward(request, response);

}

**else**

{

out.println("<div style='font-size:30px; color:red'>"+"Userid and password does not matched "+"</div>");

RequestDispatcher view =

request.getRequestDispatcher("log.jsp");

view.include(request, response);

}

}

**else**

{

out.println("<div style='font-size:30px; color:red'>"+"Please fill userid and password"+"</div>");

RequestDispatcher view=request.getRequestDispatcher("log.jsp");

view.include(request, response);

}

}**catch**(Exception e){}

**finally**{

out.close();

}

}

}

* 1. **Jsp page**

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

pageEncoding="ISO-8859-1" session="true" import="internet.\*"%>

<%

GeneratedId gen = **new** GeneratedId();

**int** form\_id = gen.generateFormId();

%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>Subscription Form</title>

<link rel="stylesheet" href="style2.css" type="text/css" />

<script language="javascript">

**function** validateForm() {

**var** Myname = document.subscribe.name.value;

**var** re=/^[a-zA-z]+$/;

**if** (Myname == "" || Myname==**null**) {

alert("Name feild cannot be blank");

document.subscribe.name.focus();

**return** **false**;

}

**else** {

**if**(Myname.match(re))

{

alert("Name entry is valid");

**return** **true**;

}

**else**{

alert("Enter valid name");

document.subscribe.name.focus();

**return** **false**;

}

}

Myname = document.subscribe.address.value;

**if** (Myname == "" || Myname==**null**) {

alert("Address feild cannot be blank");

document.subscribe.address.focus();

**return** **false**;

}

Myname=doument.subscribe.email.value;

**if** (Myname == "" || Myname==**null**) {

alert("Email field cannot be blank");

document.subscribe.email.focus();

**return** **false**;

}

Myname=document.subscribe.phone.value;

**if** (Myname == "" || Myname==**null**) {

alert("Phone field cannot be blank");

document.subscribe.phone.focus();

**return** **false**;

}

Myname=document.subscribe.dob.value;

**if** (Myname == "" || Myname==**null**) {

alert("Date of Birth field cannot be blank");

document.subscribe.dob.focus();

**return** **false**;

}

Myname=document.subscribe.proof.value;

**if** (Myname == "" || Myname==**null**) {

alert("Proof field cannot be blank");

document.subscribe.proof.focus();

**return** **false**;

}

subscribe.submit();

}

</script>

</head>

<body>

<table width="900" border="0" align="center">

<tr>

<td colspan="2"></td>

</tr>

<tr>

<td width="900" valign="top"><%@ include file="internet\_default\_menu.jsp"%></td>

</tr>

<tr>

<td width="750">

<table border="0" align="top" width=100%>

<formname="subscribe"method="post"action="SubscribeServlet">

<%

%>

<tr>

<tdbgcolor='#AAAAAA'colspan='4'align=center height=20>

<b>Contact Information</b>

</td>

</tr>

<tr>

<td width=150>Form Id</td>

<td align='left'>

<input type='text' name='form\_id'id='form\_id ' size='10' value='<%=form\_id%>' /></td>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

<tr>

<td>Name

<sup>\*</sup>

</td>

<td>

<input type='text' name='name' id='name ' size='40'value='' /></td>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

<tr>

<td>Address<sup>\*</sup></td>

<td><input type='text' name='address' id='address' size='40'value='' /></td>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

<tr>

<td>Email<sup>\*</sup></td>

<td><input type='text' name='email' id='email' size='25'value='' /></td>

</tr>

<tr>

<td>Phone<sup>\*</sup></td>

<td><input type='text' name='phone' id='phone' size='15' maxLength="10" value='' /></td>

</tr>

<tr>

<td bgcolor='#AAAAAA' colspan='4' align=center height=20><b>PersonalInformation</b></td>

</tr>

<tr>

<td colspan='4'><br> DOB<sup>\*</sup> <input type='text'name='dob' id='dob' size='10' value='' />(yyyy-mm-dd)&nbsp;&nbsp;&nbsp;

<br>

<br>

Gender&nbsp;&nbsp;&nbsp;<input type='radio'name='gender' id='gender' size='10' value='Male' /> Male

&nbsp;&nbsp;

<input type='radio' name='gender' id='gender'size='10' value='Female'> Female &nbsp;&nbsp;&nbsp;

<br>

<br>

Nationality&nbsp;&nbsp;&nbsp;<select name='nationality' id='nationality'>

<option selected>Select Nationality

<option value='IN'>Indian

<option value='RS'>Russian

<option value='PK'>Pakistani

<option value='AM'>American

<option value='BR'>British

<option value='SR'>Srilankan

</select>

<br>

</td>

<tr>

<td bgcolor='#AAAAAA' colspan='4' align=center height=20>

<b>Proof Details</b>

<td>

</tr>

<br>

<tr>

<td>Proofs</td>

<td>

Proofs&nbsp;&nbsp;&nbsp;<select name='proofs' id='proofs'>

<option selected>Select proof

<option value='PC'>Pan card

<option value='DL'>Driving licsense

<option value='AC'>Adhar card

<option value='EB'>Electricity bill

<option value='TB'>Telephone bill

</select>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

<tr>

<td>All the ( <sup>\*</sup>) marked are mandatory

</td>

</tr>

<tr>

<td colspan=4 align=center>

<form method="get" action="subscribe.html">

<input type='submit' name='submit' id='submit' size='10'value='Submit' onClick="return validateForm()" />

</form>

</td>

</tr>

</td>

</tr>

</table>

</body>

</html>

1. **TESTING DOCUMENT**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sr.No** | **TEST CASE** | **FUNCTIONALITY** | **EXPECTED**  **RESULT** | **ACTUAL RESULT** | **REMARKS** |
| 1. | Login | Login into for menu | Getting logged in | Getting logged in | Pass |
| 2. | Adding records | Insert | Inserting records | Inserting records | Pass |
| 3. | Deleting records | Search, Delete | Deleting records | Searching and Deleting records | Pass |
| 4. | Viewing records | Search, View | Viewing records | Searching and Viewing records | Pass |
| 5. | Validation for name | Check that field is not empty | Check field not empty | Check field not empty | Pass |
| 6. | Stock Availability | Stock Availability | Stock Availability | Stock Availability | Pass |

**Table 7.1: Testing Document**

**9. CONCLUSION**

The Online Mobile Shop and internet service provider has been a way of minimizing the clerical work, which is almost a routine and consumes the most precious time.

This project has been an attempt to help the user to minimize his workload along with minimizing the paper works and saving of time.

The system has been developed in a way to make it very user friendly. It provides an on-line message and an error detection and error messages every time the user needs. Any person having a little bit of window based can run this system without any pain.

Almost all the difficulties of manual shopping have been removed by this system.

**10. REFERENCES**

**10.1. Books referred:**

[1] “Java Server Programming Black Book”,

[2]H.M.Detiel, P.J.Detiel, S.E.Santry, “Advance Java2 Platform How To Program”

[3] Rojer.S.Pressman, “Software Engineering A Practitioner’s Approach”, McGraw-Hill

**10.2. Websites referred:**

[1] [www.wikipedia.org](http://www.wikipedia.org)

[2] [www.w3schools.com](http://www.w3schools.com)

[3][www.wiley.com](http://www.wiley.com/)