ASSIGNMENT NO 4

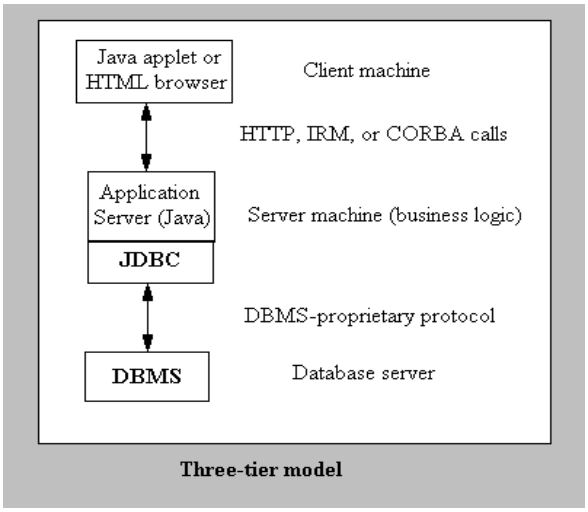
Title: Add dynamic web application features to previously selected application using Servlet, JSP and backend (MySQL / MongoDB).

Problem statement: Add dynamic web application features using Servlet, JSP and backend (MySQL / MongoDB) to any one application from the given list :

1. Online pizza order application.

Objective: To understand 3-tier applications
To explore the usage of Servlet, JSP
To understand database connectivity.

Theory:



Three-tier model

Database Server

MySQL version --- 5.5.37
Installed on Fedora 20

MySQL: Installation steps

1. On server machine open terminal.
2. change user to root user using su command.
3. type "yum install mysql mysql-server"
It will take few mins to complete.
4. on server machine type service mysqld start. (one time start,on each reboot you have to execute the command)
5. To enable mysql service to execute in any run level type chkconfig mysqld on.(this step is not
necessary)
6. Open port on server machine for mysql using command

"iptables -I INPUT -p tcp --dport 3306 -j ACCEPT"(on every server reboot)
7. execute mysql_secure_installation command.
8. It will prompt for root password. As we have not set up root password just press enter.
9. It will ask whether you want to set root password, type yes and provide new password of your choice.

Follow the remaining instructions
10. Now installation is over.
11.Type mysql -u root -p on server machine command prompt. It will prompt you for the password.
12. Once you logged in mysql you have to create separate user account and database for each student(client).
13. To create user account
type command-- > create user 'user_name'@'%' identified by 'user_password';
14. Grant permissions on the databases to users.
commands for creating database and granting permissions are given in mysql doc
For example, >grant all on Student_db.* to 'user_name'@'%";
15. The newly created account can be used by students(clients).
Client can connect to the server from any machine using following command
mysql -h ip_address_of_server_machine -u user_name_for_that_student -p
It will prompt for password.
Enter the correct password.

MySQL Client Side Installation
1. On each client machine open terminal.
2. change user to root user using su command.
3. use command --
> yum install mysql
4. Once installed client can login and connect to the server using command--
>mysql -h ip_address_of_server_machine -u user_name_for_that_student -p
For jsp development
On developers machine install suitable version of java. (java 1.7 or 1.8)
Install Apache tomcat suitable version using the steps given next. (apache tomcat 8.0.9)

Tomcat Installation Steps

1.Download tomcat tar file.
2.create tomcat installation directory anywhere using command mkdir.
e.g. mkdir /opt/tomcat_installation

3.cp apache-tomcat-{version}.tar.gz /opt/tomcat_installation
4.cd /opt/tomcat_installation
5.tar -xvf apache-tomcat-{version}.tar.gz
6.It will extract apache-tomcat-{version}.tar.gz
7.It will also include bin directory where containing binaries for tomcat.
8. cd apache-tomcat-{version}
8.cd bin.
9.To start tomcat use- ./startup.sh
It will show Tomcat started.
10.Now copy html file that you want to host on tomcat server into tomcat_installation/webapps/ROOT/
11.Open browser and type http://localhost:8080/hello.html
if hello.html is in /opt/tomcat_installation/ webapps/ROOT/ , it will show contents of html.

To host jsp pages with database connectivity
12. Create lib directory using mkdir command inside tomcat_installation/webapps/ROOT/WEB-INF/
13.Now in order to host jsp pages which will connect to database, we have to copy jstl-1.2.jar and mysql-connector.jar to tomcat_installation/webaps/ROOT/WEB-INF/lib directory. You can download this jar file from ftp server of the lab
14. Shutdown tomcat server using ./shutdown.sh command from bin directory.
15.cp jsp files to tomcat_installation/webapps/ROOT/.
16. Start tomcat using ./startup.sh
17. You can now try the sample program that follows.

Sample Application: Login
Step A: Creating table and inserting records
1. Create table users in your mysql database using :
>mysql -h ip_address_of_server_machine -u user_name_for_that_student -p
>use Student_db; /* to use your database*/
>create table users(username varchar(10), password varchar(8));
/* create table in Student_db*/
2. Insert some records into users table.
>insert into users values ('abc' , '123');
>insert into users values ('xyz' , '456');
>insert into users values ('pqr' , '789');
3. To check
> select * from users;

Step B: Creating html page

.html file – to be saved in tomcat_installation/webaps/ROOT/
Open editor and create a file login.html
Paste following lines in the login.html and save.
login.html

```
<body>
<form action="login.jsp" method="post">
User name :<input type="text" name="usr" />
password :<input type="password" name="pwd" />
<input type="submit" />
</form>
</body>
```

Step C: Developing jsp with database connectivity
.jsp file – to be saved in tomcat_installation/webaps/ROOT/
Open editor and create a file login.jsp
Paste following lines in the login.jsp and save.

```
<%@ page import ="java.sql.*" %>
<%@ page import ="javax.sql.*" %>
<%
String userid=request.getParameter("usr");
//session.putValue("userid",userid);
String pwd=request.getParameter("pwd");
Class.forName("com.mysql.jdbc.Driver");
java.sql.Connection con =
DriverManager.getConnection("jdbc:mysql://mysql_server_ip:3306/
Student_db","username‖,
"User_password");
Statement st= con.createStatement();
ResultSet rs=st.executeQuery("select * from users where
username='"+userid+"'");
if(rs.next())
{
if(rs.getString(2).equals(password))
{
out.println("welcome "+userid);
}
else
{
out.println("Invalid password try again");
}
}
%>
```

To run the application

Open the browser

Type – http://localhost:login.html

(if you are accessing login application from the same machine---developer)

OR

Type-- http://ip_address_of developer_machine:login.html

(If you are accessing login application from any other machine--- client)

HTML page contents will be shown.

Enter username and password ,

Click on submit query button.

If you provide valid username and correct password,

the WELCOME page will be shown.

Conclusion:

We successfully created dynamic web application features to previously selected application using Servlet, JSP and backend (MySQL / MongoDB).