

# Phase 2 end-Project

---

Project: User Login Using Hibernate, JSP, and Java Servlet  
Developer: Diyu Mathew

---

Duration: 3 weeks

Type: Scrum

Sprint Duration: 1 week

---

## Sprint 1:

- index page
- registration function

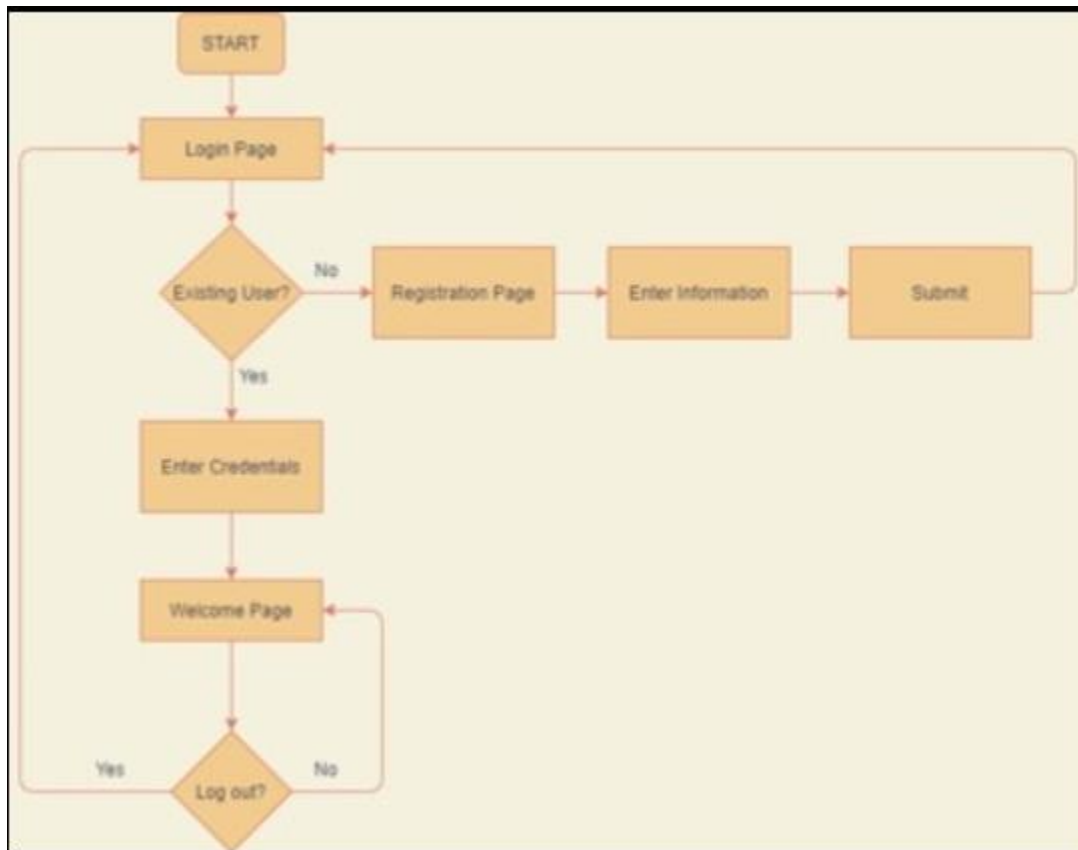
## Sprint 2:

- user login validation

## Sprint 3:

- Coding for expected or possible exceptions
- Unexpected events...

## FlowChart



## GIT HUB URL : [User Login](#)

### CODE EXPLANATION

Entity Class : UserEntity.

Uses Lombok for getters and setters

Annotated with DB table and column names

```
package com.hcl.simplilean.UserLogin.entity;

import java.io.Serializable;

@Table(name = "User")
@Entity
@Data
public class UserEntity implements Serializable {

    /**
     * Entity Class. getters and setters generated using project lombok // @Data
     * annotation
     */
    private static final long serialVersionUID = 1L;

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name = "id")
    private long id;

    @Column(name = "user_id", unique = true)
    private String userid;

    @Column(name = "first_name")
    private String firstName;

    @Column(name = "last_name")
    private String lastName;

    @Column(name = "pwd")
    private String pwd;

    @Column(name = "email", unique = true)
    private String email;
}
```

### Services Class

It contains the major operations like validate, save and Find

Find function finds the user from the db for the given user id, if exists it will return the entire object.

Validate function receives the userid and pwd from user through jsp via java servlet and searches the db using Find() and gets the user object if exists. Then matches the input pwd and pwd from db.

Save () saves the User object to db using Entity class and hibernate.

**Project code.**

```
package com.hcl.simplilean.UserLogin.entity;
```

```
import java.io.Serializable;
```

```
import javax.persistence.Column;
```

```
import javax.persistence.Entity;
```

```
import javax.persistence.GeneratedValue;
```

```
import javax.persistence.GenerationType;
```

```
import javax.persistence.Id;
```

```
import javax.persistence.Table;
```

```
import lombok.Data;
```

```
@Table(name = "User")
```

```
@Entity
```

```
@Data
```

```
public class UserEntity implements Serializable {
```

```
    /**
```

```
     * Entity Class. getters and setters generated using project lombok //@Data
```

```
     * annotation
```

```
     *
```

```
     */
```

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

```
    @Id
```

```
    @GeneratedValue(strategy = GenerationType.IDENTITY)
```

```
@Column(name = "id")
private long id;

@Column(name = "user_id", unique = true)

private String userid;

@Column(name = "first_name")

private String firstName;

@Column(name = "last_name")

private String lastName;

@Column(name = "pwd")

private String pwd;

@Column(name = "email", unique = true)

private String email;

}

package com.hcl.simplilean.UserLogin.hibernate;

import org.hibernate.SessionFactory;
import org.hibernate.boot.registry.StandardServiceRegistryBuilder;
import org.hibernate.cfg.Configuration;
```

```
import org.hibernate.service.ServiceRegistry;
```

```
import com.hcl.simplilean.UserLogin.entity.UserEntity;
```

```
public class HibernateUtils {
```

```
    static SessionFactory hibernateSessionFactory;
```

```
    public static SessionFactory buildSessionFactory() {
```

```
        /* Telling hibernate where to connect to + what entities are supported */
```

```
        Configuration conf = new Configuration();
```

```
        conf.addAnnotatedClass(UserEntity.class);
```

```
        conf.configure("hibernate.cfg.xml");
```

```
        ServiceRegistry registry = new StandardServiceRegistryBuilder()
```

```
            .applySettings(
```

```
                conf.getProperties()
```

```
            ).build();
```

```
        hibernateSessionFactory = conf.buildSessionFactory(registry);
```

```
        return hibernateSessionFactory;
```

```
    }
```

```
}
```

```
package com.hcl.simplilean.UserLogin.services;
```

```
import org.hibernate.Session;
```

```

import com.hcl.simplilean.UserLogin.entity.UserEntity;
import com.hcl.simplilean.UserLogin.hibernate.HibernateUtils;

public class Services {

    public static Session hibernateSession;

    // Function to Search Database for existing elements with given userid
    public static UserEntity find(String userid) {
        hibernateSession = HibernateUtils.buildSessionFactory().openSession();
        hibernateSession.beginTransaction();

        //Hibernate query using entity parameters class to map to the database
        UserEntity u = (UserEntity) hibernateSession.createQuery("FROM UserEntity U WHERE
U.userid= :userid")

                .setParameter("userid", userid).uniqueResult();

        //returns User information UserEntity
        return u;
    }

    //Validates the userid and password given by the user
    public static boolean validate(String userid, String pwd) {

        UserEntity u = Services.find(userid);
        if (u != null) {

            // comparison btw password from DB and User

```

```

        if (u.getPwd().equals(pwd)) {
            return true;
        }

    }

    return false;

}

// Saves UserEntity read from user to Database
public static void save(UserEntity U) {
    hibernateSession.saveOrUpdate(U);
    hibernateSession.getTransaction().commit();
}

}

package com.hcl.simplilean.UserLogin.servlets;

import java.io.IOException;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import com.hcl.simplilean.UserLogin.entity.UserEntity;

```

```

import com.hcl.simplilean.UserLogin.services.Services;

/**
 * Servlet implementation class login
 */

//url mapping
@WebServlet("/loginservice")
public class LoginService extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public LoginService() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
        response.setContentType("text/html");

        if
(Services.validate(request.getParameter("userid"),request.getParameter("password"))){
            //if login credentials match , forward to welcome.jsp.

            UserEntity u=Services.find(request.getParameter("userid"));

```



```

        RequestDispatcher rd =
getServletContext().getRequestDispatcher("/welcome.jsp");

        request.setAttribute( "first_name",u.getFirstName());
        request.setAttribute( "last_name",u.getLastName());
        request.setAttribute( "userid",u.getUserid());
        request.setAttribute( "email",u.getEmail());
        rd.forward(request, response);
    }else {

        //if login credentials doesn't match give error message
        RequestDispatcher rd = getServletContext().getRequestDispatcher("/index.jsp");
        request.setAttribute( "msg","<div style=\"color:Tomato;\">Wrong Username or
Password used</div>");
        rd.forward(request, response);
    }
}

}

package com.hcl.simplilean.UserLogin.servlets;

import java.io.IOException;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import com.hcl.simplilean.UserLogin.entity.UserEntity;
import com.hcl.simplilean.UserLogin.services.Services;

```

```

/**
 * Servlet implementation class RegisterService
 */
//Servlet mapping
@WebServlet("/RegisterService")
public class RegisterService extends HttpServlet {

    private static final long serialVersionUID = 1L;
    // static Session hibernateSession;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public RegisterService() {
        super();
    }

    public void destroy() {
        Services.hibernateSession.close();
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
     *      response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        // checks whether password and confirm password match or not
        if (request.getParameter("pwd").equals(request.getParameter("cnfpwd"))) {

```

```

// if passwords match checks for existing user with same user id

UserEntity user = Services.find(request.getParameter("userid"));
if (user == null) {
    // creating new user
    user = new UserEntity();
    user.setUserId(request.getParameter("userid"));
    user.setFirstName(request.getParameter("first_name"));
    user.setLastName(request.getParameter("last_name"));
    user.setPwd(request.getParameter("pwd"));
    user.setEmail(request.getParameter("email"));
    Services.save(user);

    RequestDispatcher rd =
getServletContext().getRequestDispatcher("/index.jsp");
    request.setAttribute("msg", "<div style=\"color: DodgerBlue;\">" +
"Registraion Successfull !!</div>");
    rd.forward(request, response);

} else {
    // error message is userid already exists
    RequestDispatcher rd =
getServletContext().getRequestDispatcher("/registration.jsp");
    request.setAttribute("msg", "<div style=\"color: Tomato;\">" +
"Registraion UnSuccessfull !!"
+ " Username already exists<br>" + "Please use
different Username <br></div>");
    rd.forward(request, response);
}
} else {
    // error message if password and confirm password doesn't match

```

```

        RequestDispatcher rd =
getServletContext().getRequestDispatcher("/registration.jsp");

        request.setAttribute("msg", "<div style=\"color:Tomato;\">" + "Registraion
UnSuccessfull !!"

        + " Passwords Doesn't match<br></div>");

        rd.forward(request, response);

```

```

    }

}

}

```

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Login</title>
</head>
<body>
    <div align="center">
        <h2 >Log In</h2>

        ${msg}
        <form action="loginservice"
            method="post">
            <table style="width: 50%">
                <tr>
                    <td>User Name:</td>
                    <td><input type="text" name="userid" /></td>
                </tr>

```

```

        <tr>

            <td>Password:</td>

            <td><input type="password" name="password" /></td>

        </tr>

    </table>

    <input type="submit" value="login" />

</form>

<h5>*****</h5>

<h4>New Here...</h4>

<a href="/user/registration.jsp">Sign Up</a>

</div>

</body>

</html>

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

<!DOCTYPE html>

<html>

<head>

<meta charset="ISO-8859-1">

<title>Sign Up</title>

</head>

<body>

    <div align="center">

        <h2>User Registration</h2>

        <form action="RegisterService"
            method="post">

            <table style="width: 50%">

                <tr>

                    <td>First Name</td>

```

```
        <td><input type="text" name="first_name" /></td>
    </tr>
    <tr>
        <td>Last Name</td>
        <td><input type="text" name="last_name" /></td>
    </tr>
    <tr>
        <td>UserName</td>
        <td><input type="text" name="userid" /></td>
    </tr>
    <tr>
        <td>Password</td>
        <td><input type="password" name="pwd" /></td>
    </tr>
    <tr>
        <td>Confirm Password</td>
        <td><input type="password" name="cnfpwd" /></td>
    </tr>
    <tr>
        <td>Email Address</td>
        <td><input type="text" name="email" /></td>
    </tr>
</table>
<input type="submit" value="Submit" />
</form>
<h5>*****</h5>
${msg}
<h4>Already have an account?</h4>
<a href="/user/index.jsp">Log In Here</a>
</div>
```

```
</body>
</html>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>welcome</title>
</head>
<body>
<div align = "center">
<h3>Welcome</h3><br>
        <table style="width: 50%">
            <tr>
                <td>Name</td>
                <td> ${first_name} ${last_name}</td>
            </tr>
            <tr>
                <td>UserName</td>
                <td> ${userid}</td>
            </tr>
            <tr>
                <td>Email Address</td>
                <td> ${email}</td>
            </tr>
        </table>
        <a href="index.jsp"><input type="submit" value="Log Out" /></a>
    </div>
```

</body>

</html>