

Universidad Autónoma de Nuevo León  
Facultad de Ciencias Físico Matemáticas

Laboratorio de Diseño Orientado a Objetos  
Semana #5 MVC

Profesor:

Miguel Angel Salazar

Estudiante:

Angel Adolfo Pacheco Mazuca

1656991

Dentro de esta practica se creo un login simple con el modelo MVC el cual separa todo el contenido de un proyecto por partes, tenemos el lado del cliente, la parte logica y la parte de la interface.

Empezamos borrando el html que tiene por defecto y creamos un jsp que es muy similar al html y dentro de el creamos un form con la siguiente estructura.

```
<form name="Formulario" action="LoginController" method="post">
  <label>Usuario: </label>
  <br>
  <input type="text" id="user" name="txt-username" />
  <br>
  <br>
  <label>Contraseña: </label>
  <br>
  <input type="password" id="pass" name="txt-password" />
  <br>
  <br>
  <input type="submit" id="sent" name="btn-enviar" value="Envar" />

</form>
```

Después de esto creamos un servlet como los que hemos visto hasta ahora solo que en esta ocasión borraremos todo lo que se encuentra dentro del método `processRequest` en el cual modificamos y debe quedar de esta forma.

```
//RequestDispatcher será la clase que nos apoyará a devolver
//el valor hacia la vista
RequestDispatcher dispatcher = null;

//Se recupera la información del request
String txtUsername = request.getParameter("txt-username");
String txtPassword = request.getParameter("txt-password");

//Invoca al método authenticate para validar el usuario
boolean isValidUser = Authentication.authenticate(txtUsername, txtPassword);

if(isValidUser) {

    //Construye instancia del modelo User
    User user = new User(txtUsername, txtPassword);

    //Construye parámetro para enviar a la vista success.jsp
    request.setAttribute("username", user.getUsername());

    //Se define a donde se enviará el objeto request y response.
    dispatcher = request.getRequestDispatcher("success.jsp");
    dispatcher.forward(request, response);
}
else {
    //Si el usuario es inválido, re direccionamos la petición hacia error.jsp
    response.sendRedirect("error.jsp");
}
```

Dentro de nuestro proyecto crearemos una clase de java con el nombre de `User` el cual deberá contener unos strings como se muestra en la siguiente imagen.

```
private String username;
private String password;

public User(String username, String password) {
    this.username = username;
    this.password = password;
}

public String getUsername() {
    return this.username;
}
```

De la misma manera debemos de crear otra clase la cual nos servirá para autenticar el pequeño login que habíamos hecho en el jsp

```
//Variables *dummy*.
//Se hardcodean los valores "Miguel" y "MiPassword" para emular que se obtuvieron de una base de datos
String userDataBase = "Miguel";
String passwordDataBase = "MiPassword";

//username.equals(userDataBase) realiza una comparación entre las cadenas username y userDataBase
//Si son iguales devuelve true. Si son diferentes devuelve false.
if(username.equals(userDataBase) && password.equals(passwordDataBase)) {
    return true;
}
else {
    return false;
}
```

Deberá quedar como se muestra en la imagen,

Para poder hacer que todo nuestro modelo funcione debemos importar uas clases a nuestro servlet

```
import javax.servlet.RequestDispatcher;
```

```
import week5.models.User;
```

```
import week5.models.Authentication;
```

Despues debemos crear otro jsp con el nombre de success.jsp y dentro de el deberá tener la estructura siguiente:

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Inicio de sesión válido</title>
    <link rel="stylesheet" type="text/css" href="style.css">
  </head>
  <body>

    <h1>Inicio de sesión válido</h1>

    <h2>Hola <%= request.getAttribute("username") %></h2>
  </body>
</html>
```

Y tendremos que crear otro jsp con el nombre de error.jsp el cual deberá tener la siguiente forma en su estructura:

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Inicio de sesión inválido</title>
    <link rel="stylesheet" type="text/css" href="style.css">
  </head>
  <body>
    <header>
      <h1>
        Usuario o contraseña incorrecta.
      </h1>
      <p>
        Intentalo de nuevo, solo da click al siguiente enlace: <a href="login.jsp">Regresar a login</a>
      </p>
    </header>
  </body>
</html>
```

Para probar la aplicacion debemos de colocar en la barra de direcciones la siguiente liga: <http://localhost:8080/login-mvc/login.jsp>

En este caso es login-mvc por el nombre del proyecto puede cambiar según su nombre del proyecto.

## Bienvenido, por favor inicia sesion.

Usuario:

Contraseña:

Enviar

Tendremos algo parecid a esto, y debemos poner en este caso "Miguel" en user y en contraseña "MiPassword"

Si es introducido de manera correcta deberá aparecer la siguiente pagina:

## **Inicio de sesión válido**

**Hola Miguel**

En caso de que se haya introducido de manera incorrecta asi se veria nuestra respuesta:

## **Usuario o contraseña incorrecta.**

Intentalo de nuevo, solo da click al siguiente enlace: [Regresar a login](#)

Esta es una manera diferente a lo que hemos estado viendo dentro de los servlets para responder a diferentes acciones que puede tener nuestra aplicación la cual a mi punto de vista se me hace mas ordenada y mas estructurada que estar inyectando código html dentro de un archivo vacio.