



Nombre: ALEXIS DURAN TIJERINA		Matrícula: 1479832
Unidad de Aprendizaje: LDOO	Nombre del profesor: Miguel Salazar	
Módulo: PRACTICA 3	Actividad: Servlet	
Fecha: 20/09/2017		
<i>Crear un login simple con MVC</i>		

El objetivo es construir un sistema de autenticación simple basado en el patrón de diseño MVC (Model-View-Controller).

Primero se hacen las construcciones de vistas con un simple proyecto en java web eliminando el index y creando 3 archivos .JSP (login.jsp, error.jsp, success.jsp)

En el login.jsp se añade un código para una tipo bienvenida creando un form formulario que ejecutara una acción tipo servlet, action = LoginController

El formulario tendrá inputs de tipo post para enviar el texto a este servlet Quedando de la siguiente manera el código html:

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Inicio de Sesión</title>
  </head>
  <body>
    <h1>Bienvenido</h1>

    <form action="LoginController" method="post">
      Usuario: <input name="usuario" type="text"><br>
      contraseña: <input name="contrasena" type="password"><br>
      <input type="submit" value="Iniciar sesion"><br>
    </form>

  </body>
</html>
```

El siguiente archivo error quedaría con un mensaje de error y un href para que pueda regresar a login.jsp:

```
<body>
  <h1>Usuario o contraseña incorrectos </h1>
  <a href="login.jsp">Regresar</a>
</body>
```

El archivo success.jsp se modificara en los siguientes laboratorios

- Construcción de controladores

El primer controlador seria el servlet LoginController en el paquete week5.controllers

En este paso solo se agregan las variables de usuarios y contraseñas.

```
String txtUser = request.getParameter("usuario");
String txtPass = request.getParameter("contrasena");
```

#### -Construcción de modelos

Se crea una clase en java de nombre User que funcionara como modelo en el paquete week5.models

Se ingresara código para crear una tipo variable que solo hara la función de una tipo base de datos para hacer una autenticación.

Seria el siguiente código:

```
public class Authentication {

    public static boolean authenticate(String username, String password) {

        //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;
        }
    }
}
```



Para hacer valida la valoración del usuario y contraseña se necesita que en nuestro controlador se importen las clases `week5.models.*` y la clase de `RequestDispatcher` dentro del paquete `javax.servlet`:

```
import javax.servlet.RequestDispatcher;

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

En el controlador el código quedaría de la siguiente manera:

```
protected void processRequest(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {

    //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, reedireccionamos la petición hacia
error.jsp
        response.sendRedirect("error.jsp");
    }
}
```

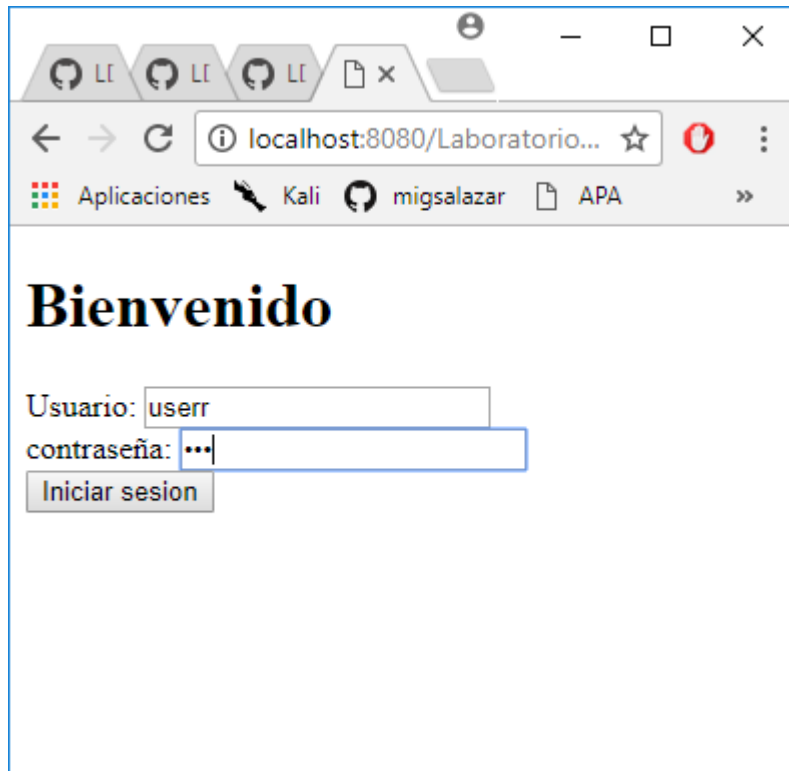
Ya para finalizar el archivo `success.jsp` se modifica y se ingresa el siguiente código para que muestre un mensaje y el usuario que se ingreso en el `login.jsp`

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>Inicio de sesión válido</title>
  </head>
  <body>

    <h1>Inicio de sesión válido</h1>

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

## RESULTADOS



**Bienvenido**

Usuario:

contraseña:

# FCFM

FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS

