



Nos ha quedado un tema pendiente en el PDF anterior.



¿Que debemos hacer para comprobar la contraseña?

Login

Introduzca su nombre:  Introduzca su contraseña:

Estado	Método	Dominio	Archivo	Iniciador	Tipo	Transferido	Tamaño	0 ms
200	GET	localhost:8080	/	document	html	458 B	315 B	5 ms

Login

localhost:8080/login.do

Hola JoseRa todavía no sabemos que hacer con MicontraseÃ±a

Estado	Método	Dominio	Archivo	Iniciador
200	POST	localhost:8080	login.do	document



Y la pregunta es: ¿Como validamos el usuario?



# UD 1: Introducción a los lenguajes de servidor

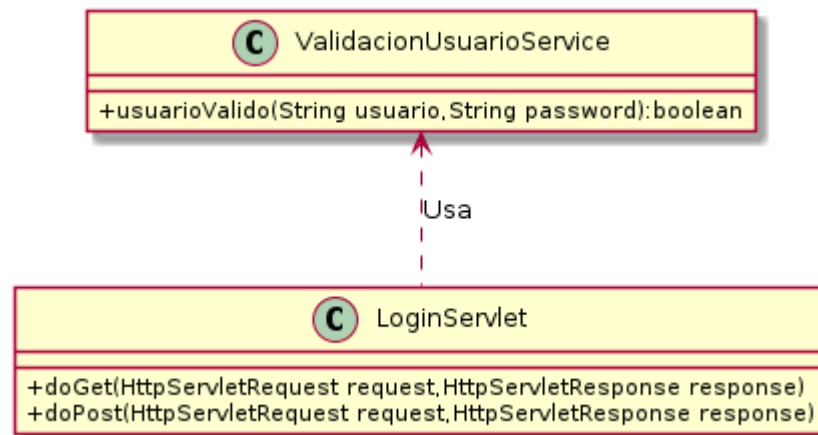
## 4.- JEE: Servlets, JSP y JSTL



Desarrollo Web en Entorno Servidor - Joseramon.profesor@gmail.com

### Servicios Java:

- Para validar si el usuario es válido o no crearemos nuestro **primer servicio**.
- Un servicio es una clase Java que se **proporciona un servicio/funcionalidad al resto de clases de nuestra aplicación**, en nuestro caso el servicio de validación de usuario proporcionará un servicio a la clase LoginServlet.java.





# UD 1: Introducción a los lenguajes de servidor

## 4.- JEE: Servlets, JSP y JSTL



Desarrollo Web en Entorno Servidor - Joseramon.profesor@gmail.com

### ... continuación **Servicios Java:**

- Antes de comenzar, como vamos a crear más ficheros java, pongamos un ***nombre al package más apropiado*** (actualmente “webapp”). Si nuestra aplicación tenía como Group Id “org.alumno.nombreAlumno” vamos a ***cambiar el nombre del package a “org.alumno.nombreAlumno.primer\_je” en LoginServlet.java*** y a partir de ahora dejaremos las nuevas clases colgando de este package.

- ***Creemos una nueva clase ValidacionUsuarioService.java:***

```
1 package org.profesor.joseramon.primer_je;
2
3 public class ValidacionUsuarioService {
4     public boolean usuarioValido(String usuario,String password) {
5         //En condiciones normales se accedería a una base de datos para
6         //confirmar la contraseña, pero por simplicidad lo validamos
7         //comprobando si son unos valores fijos.
8         //El alumno debe cambiar los valores del usuario y contraseña validos
9         if (usuario.contentEquals("joseramon") && password.contentEquals("miContraseña"))
10             return true;
11         return false;
12     }
13 }
```





... continuación **Servicios Java:**

- Ahora *añadimos el servicio a nuestro Servlet para poder utilizarlo y comprobamos que funciona correctamente probándolo desde el navegador:*

Project Explorer

mi-primera-webapp-jee

JAX-WS Web Services
Java Resources

src/main/java

org.profesor.joseramon.primer\_je

LoginServlet.java
ValidacionUsuarioService.java
src/main/resources
src/test/java
src/test/resources
Libraries
Deployed Resources
src

main

java

org

profesor

joseramon

primer\_je

LoginServlet.java

```

1 package org.profesor.joseramon.primer_je;
2
3 import java.io.IOException;
4
11
13 * Browser sends Http Request to Web Server
20
21 /*Java Platform, Enterprise Edition (Java EE) JEE8
33 @WebServlet(urlPatterns = "/login.do")
34 public class LoginServlet extends HttpServlet {
35     ValidacionUsuarioService servicio = new ValidacionUsuarioService();
36
37 @Override
38 protected void doGet(HttpServletRequest request,
39     HttpServletResponse response) throws IOException, ServletException {
40     request.getRequestDispatcher("/WEB-INF/views/login.jsp").forward(request,
41     }
42 @Override
43 protected void doPost(HttpServletRequest request,
44     HttpServletResponse response) throws IOException, ServletException {
45     //El alumno debe leer el nombre y el password , llamar
46     //al servicio para comprobar si el usuario es válido.
47
48     //Si es válido deberá ir a la página de bienvenida
49
50     //Si no es válido deberá volver al login
51 }
52 }

```



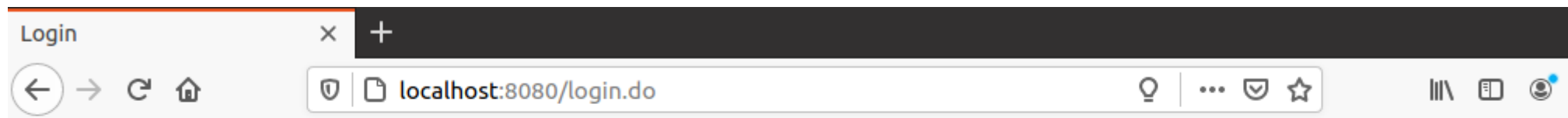
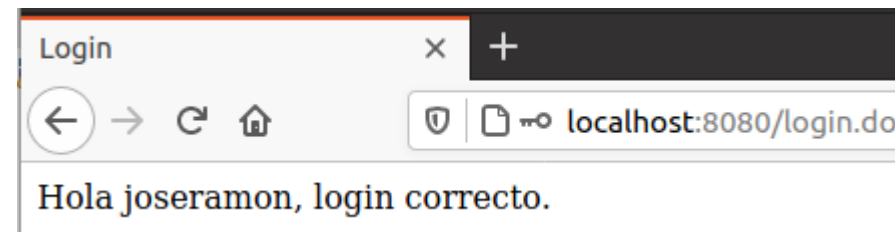
### ... continuación Servicios Java:

- Cuando nos equivocamos nos vuelve a pedir usuario y contraseña como si fuera la primera vez que nos logeamos. Sería muy conveniente añadir un atributo “errores” que puede pasarse a login.jsp de manera que si hay errores se muestren. *Realiza los cambios necesarios en LoginServlet.java, login.jsp y bienvenida.jsp para tener el siguiente resultado:*

```

1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6 <meta charset="UTF-8">
7 <title>Login</title>
8 </head>
9 <body>
10 <p><font color="red">${errores}</font></p>
11 <form action="/login.do" method="post">
12   Introduzca su nombre: <input type="text" name="nombre"/>
13   Introduzca su contraseña: <input name="password" type="password" />
14   <input type="submit" value="Login">
15 </form>
16 </body>
17 </html>
18

```



Usuario 'Pedro' o contraseña incorrecta

Introduzca su nombre:  Introduzca su contraseña:



# UD 1: Introducción a los lenguajes de servidor

## 4.- JEE: Servlets, JSP y JSTL



Desarrollo Web en Entorno Servidor - Joseramon.profesor@gmail.com

### EJERCICIO:

Sigue todos los pasos de los PDF y sube la aplicación final al moodle.

Para ello:

1º Haz un “Run As \Maven Clean” para dejar solo los fichero fuentes y quitar momentaneamente los necesarios para ejecutar la aplicación (dependencias).

2º Comprime la carpeta de tu aplicación y ponle como nombre al fichero comprimido UD1\_practica4\_nombreAlumno.tar.gz donde nombreAlumno es el nombre del alumno que entrega la práctica.

3º Súbela al moodle.

**IMPORTANTE:** No comprimir en RAR, porque Ubuntu no lo lee bien y en clase tenemos Ubuntu. Si tuviesemos Windows, podemos comprimir en ZIP.