

Branch: master ▾ LDOO / lab05 /

[Create new file](#) [Upload files](#) [Find file](#) [History](#)

migsalazar lab 05

Latest commit 080a218 10 days ago

[README.md](#)

lab 05

10 days ago

README.md

Laboratorio 5

Manejo de sesiones

Objetivo: Al finalizar las actividades de este laboratorio, deberás ser capaz de comprender los aspectos básicos del trabajo de sesiones en Java EE.

Preparación

La descripción de este laboratorio parte del supuesto que ya se cuenta con la versión de NetBeans (EE) y que incluye un contenedor Web GlassFish o Apache Tomcat atendiendo un puerto de escucha 8080.

Actividades

En este laboratorio se realizará una mini-aplicación web que funcione con el patrón MVC y haga uso de sesiones. El objetivo es construir un login sencillo y almacenar la información del usuario en variables de sesión.

Actividad 1 - Ajuste de proyecto de Laboratorio 4

- 1.- Crear un proyecto de tipo Java Web en NetBeans de nombre `Laboratorio5`.
- 2.- Elimina el archivo `index.html` que crea por defecto NetBeans y que se encuentra dentro de la carpeta `Web Pages`.
- 3.- Utiliza los archivos del `Laboratorio4`:
 - Copia y pega los archivos `login.jsp` y `error.jsp` del proyecto `Laboratorio4`.
 - Crea un nuevo servlet de nombre `LoginController` dentro del paquete `Laboratorio5.controllers` y agrega el código del controlador del `Laboratorio4`. En este punto, asegúrate de que se ha creado tu archivo `web.xml` dentro de la carpeta `WEB-INF`.
 - De la misma manera que el punto anterior, utiliza los modelos creados en el proyecto `Laboratorio4`. No olvides crear el paquete de los modelos.
- 4.- Agrega un nuevo archivo de nombre `profile.jsp` dentro de `Web Pages`.

Actividad 2 - Modificación de modelo

El modelo `user` original contiene dos propiedades `username` y `password`. Deberás realizar los siguientes cambios:

- Agregar una nueva propiedad `Nombre` de tipo `String` y privada.
- Agregar una nueva propiedad `Apellidos` de tipo `String` y privada.
- Ambas propiedades tendrán información `dummy` como se creó en el `Laboratorio4`.

- Crear tres métodos: `String getName()`, `String getLastName()` y `String getFullName()`. Estos métodos deberán devolver el nombre, apellidos y el nombre completo (nombre + apellidos). El método `getFullName()` deberá hacer uso de los dos primeros métodos para su funcionamiento.

Actividad 3 - Modificación de controlador

El controlador `LoginController` original, en su método `processRequest`, realizaba una validación del usuario. De ser correcta, enviaba a la página `success.jsp`; de ser falsa, enviaba a `error.jsp`. Lo anterior se realizaba mediante el objeto `RequestDispatcher`. Para esta actividad lo dejaremos de utilizar e incluiremos nuevo código para el trabajo de sesiones:

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

    //El objeto request obtiene la sesión de la aplicación a través de getSession()
    //Se almacena la sesión en el objeto session de tipo HttpSession
    HttpSession session = request.getSession();

    String txtUsername = request.getParameter("txt-username");
    String txtPassword = request.getParameter("txt-password");

    boolean isValidUser = Authentication.authenticate(txtUsername, txtPassword);

    if(isValidUser) {
        User user = new User(txtUsername, txtPassword);

        //Establecemos variables de sesión
        session.setAttribute("username", user.getUsername());
        session.setAttribute("name", user.getName());
        session.setAttribute("fullname", user.getFullName());

        //Mostramos el perfil del usuario
        response.sendRedirect("profile.jsp");
    }
    else {
        response.sendRedirect("error.jsp");
    }
}
```

Actividad 4 - Construcción de vista Perfil

La página `profile.jsp` deberá aparentar el perfil del usuario que está iniciando sesión.

- Deberá mostrar un mensaje de bienvenida al usuario similar a la página `success.jsp` del Laboratorio4.
- Además, debemos agregar elementos de HTML para mostrar el nombre completo y el usuario.
- Validaremos la sesión del usuario al iniciar la página. Supongamos que alguien intenta tener acceso directamente a esta página tecleando en la url. Se deberá validar que puede entrar a la página si su sesión ya esta creada.

```
<%
    if(session.getAttribute("username") == null)
        response.sendRedirect("login.jsp");
%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>Esto solo es un ejemplo</title>
    </head>
    <body>
        <h1>Hola, <%= session.getAttribute("name") %></h1>
    </body>
</html>
```

Actividad 5 - Modificación de vista Login.

Supongamos que alguien ya inició sesión, si entrara de nuevo a la página de login ¿Tiene sentido pedir nuevamente las credenciales? *Of course, not my friend.*

- Validar mediante la sesión si debe o no presentarse el formulario de `login.jsp`.

```
<%  
    if(session.getAttribute("username") != null)  
        response.sendRedirect("profile.jsp");  
%>
```

Pruebas

Realiza las pruebas para todos los escenarios.

- Inicio de sesión inválido (usuario o password erróneo)
- Inicio válido (usuario o password correctos)
- Entrar directamente a `profile.jsp` con sesión y sin sesión.
- Entrar directamente a `login.jsp` con sesión y sin sesión.

Preguntas

- ¿Identificas alguna ventaja de usar sesiones en comparación al Laboratorio 4? ¿Cuáles?
- Si requerimos más información del usuario, ¿Consideras que debe seguirse trabajando con sesiones?
- ¿Qué ventajas tiene pasar información con sesión y no mediante valores en el request como en el Laboratorio 4?

Especificaciones del Reporte

- El reporte debe incluir una portada con tus datos al inicio.
- El reporte debe contener la descripción de los pasos realizados para llevar a cabo la práctica del laboratorio. Cada paso debe contar con un fragmento de código o imagen que ilustre lo descrito. Piense en el reporte como una explicación para alguien ajeno al tema y detalle los puntos técnicos que sean necesarios.
- Contesta las preguntas mencionadas en la sección anterior.
- Comprime en un archivo `.zip` el directorio raíz de la práctica.
- El envío de la práctica debe incluir dos archivos: El reporte en `PDF` y el archivo `.zip` con el código fuente del proyecto.