

## Java EE Ejercicio (MVC Parte 5)



**{"PrOgr4m4  
tu futurO"};**

Edición Back-End JAVA

## Ejercicio de Java EE. MVC y Sesión

### Ejercicio 5 de Modelo Vista Controlador y Sesión.

Como en ejercicios anteriores, imaginemos que trabajas en una pequeña emisora musical de radio local que quiere dar otro servicio a sus oyentes, y para ello crea una pequeña web donde el usuario pueda logarse y ver el listado de los diferentes eventos musicales que hay.

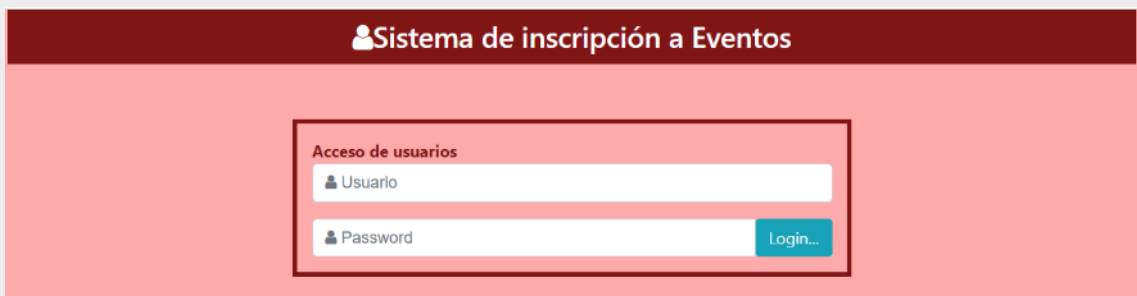
En este ejercicio, vamos a seguir trabajando con el patrón Modelo Vista Controlador, además del uso de sesiones y con distintos controlladores.

Nos debemos **crear un nuevo proyecto web “JavaEEGestionEventosEmisora”**, y dentro de este nuevo proyecto vamos a copiar y crear aquellos componentes o recursos que sean necesarios para que la aplicación funcione correctamente.

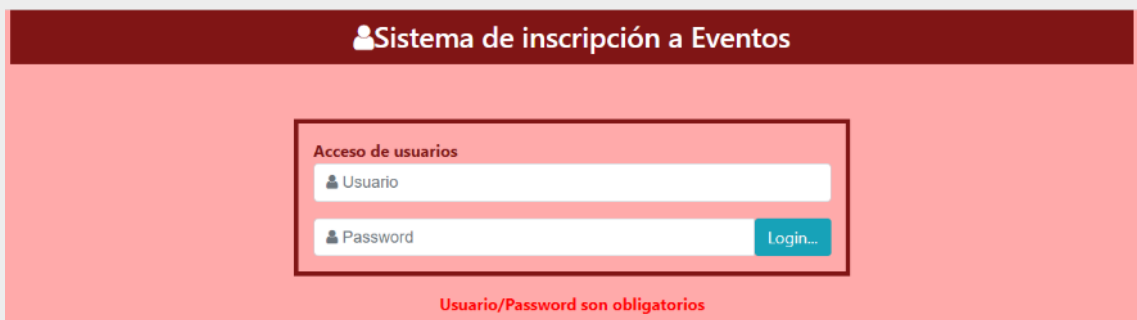
Para este nuevo ejercicio, vamos a usar una **nueva BBDD llamada “festivaldb2”, para lo cual, junto a este documento se proporciona el fichero “festivaldb2.sql” con el script necesario para crear dicha BBDD, simplemente tendrás que abrir el script en la herramienta de Workbench y ejecutarlo.**

La aplicación Web resultante debe funcionar de la siguiente forma:

- Nuestra aplicación arranca con la página inicial (**Login.jsp**) desde la cual podremos logarnos con usuario y contraseña, de la siguiente forma.



- Si no introducimos un usuario y/o contraseña, mostrará el siguiente mensaje:



- Si introducimos un usuario y/o contraseña incorrecta, mostrará el siguiente mensaje:

The screenshot shows a login form titled "Sistema de inscripción a Eventos". It has a header bar with the title and a user icon. Below the header, there's a box labeled "Acceso de usuarios" containing two input fields: "Usuario" and "Password", followed by a "Login..." button. Below the login box, a red error message reads "Usuario/Password incorrecta".

- Si introducimos un usuario y contraseña correctos, debemos pasar a la siguiente página (**BuscarEventos.jsp**) donde se muestra la información del usuario, además podremos realizar la búsqueda de Eventos, de la siguiente forma:

The screenshot shows the user profile page titled "Sistema de inscripción a Eventos". It has a header bar with the title and a user icon. On the left, there's a link "Cerrar sesión". The main content area shows user details: "Nombre: Rebeca Velasco", "Email: rvelasco@accenture.com", "Dirección: La Finca Training", and "Teléfono: 341122334455". Below this is a section titled "Lista de Eventos" with a search input field labeled "Nombre/Descripción del evento" and a "Buscar..." button.


Desde esta página pueden ocurrir varias cosas:

- Si pulsamos "Cerrar sesión" → Se cierra la sesión y volvemos a la página de inicio para volver a logarnos.
- Si se pulsa el botón "Buscar..." del cuadro de "Lista de Eventos" y el campo "Nombre/Descripción del evento" está vacío → Muestra todos los eventos que tenemos disponibles en la aplicación, de la siguiente forma:

The screenshot shows the event list page titled "Sistema de inscripción a Eventos". It has a header bar with the title and a user icon. On the left, there's a link "Volver". The main content area is titled "LISTADO DE EVENTOS" and contains a table with 7 columns: Id, Nombre, Descripción, Lugar, Duración, Tipo, and Asientos disponibles. The table lists 9 events.

Id	Nombre	Descripción	Lugar	Duración	Tipo	Asientos disponibles
1	Concierto Metalica	World Tour 2024	Madrid	3 horas	Música	67
2	Concierto ACDC	Hellfest 2024	Francia	3 horas	Música	130
3	Concierto Iron Maiden	World Tour 2024	Australia	2 horas	Música	310
4	Concierto Guns N Roses	Concierto Gira 2023	Denver	2 horas	Música	400
5	Concierto WarCry	Daimon Tour 2023	Madrid	3 horas	Música	20
6	Concierto Tierra Santa	Presentación nuevo album	Madrid	2 horas	Música	40
7	Concierto Baron Rojo	XI Zombie Jaialdia Fest C.C.	Donostia	2 horas	Música	200
8	Concierto Mago de Oz	Tour 35 Aniversario	Los Angeles	2 horas	Música	500
9	Concierto Medina Azahara	Gira LLevo el día	Cadiz	3 horas	Música	10

- Si se pulsa el botón “Buscar...” del cuadro de “Lista de Eventos” y el campo “Nombre/Descripción del evento” está informado → Debe mostrar aquellos eventos que contengan el valor introducido en su nombre o descripción, de la siguiente forma:


 Sistema de inscripción a Eventos

[Cerrar sesión](#)

Nombre: Rebeca Velasco  
Email: rvelasco@accenture.com  
Dirección: La Finca Training  
Teléfono: 341122334455


Lista de Eventos

Buscar...

 Sistema de inscripción a Eventos

[Volver](#)

LISTADO DE EVENTOS						
Id	Nombre	Descripción	Lugar	Duración	Tipo	Asientos disponibles
5	Concierto WarCry	Daimon Tour 2023	Madrid	3 horas	Música	20

 Sistema de inscripción a Eventos

[Cerrar sesión](#)

Nombre: Rebeca Velasco  
Email: rvelasco@accenture.com  
Dirección: La Finca Training  
Teléfono: 341122334455

Lista de Eventos

Buscar...

 Sistema de inscripción a Eventos

[Volver](#)

LISTADO DE EVENTOS						
Id	Nombre	Descripción	Lugar	Duración	Tipo	Asientos disponibles
1	Concierto Metalica	World Tour 2024	Madrid	3 horas	Música	67
3	Concierto Iron Maiden	World Tour 2024	Australia	2 horas	Música	310
5	Concierto WarCry	Daimon Tour 2023	Madrid	3 horas	Música	20
8	Concierto Mago de Oz	Tour 35 Aniversario	Los Angeles	2 horas	Música	500

- Si se pulsa el botón “Buscar...” del cuadro de “Lista de Eventos y el campo “Nombre/Descripción del evento” está informado, pero no hay eventos que cumplan con dicho criterio → Debe mostrar la lista vacía y el siguiente mensaje de error, de la siguiente forma:

The screenshot shows a web interface titled "Sistema de inscripción a Eventos". At the top left, there is a link "Cerrar sesión". Below it, a box displays user information: "Nombre: Rebeca Velasco", "Email: rvelasco@accenture.com", "Dirección: La Finca Training", and "Teléfono: 341122334455". Below this, another box titled "Lista de Eventos" contains a search input field with the text "dfasdfas" and a blue button labeled "Buscar...".

The screenshot shows the same web interface as above. The search input field in the "Lista de Eventos" box now contains the placeholder text "Nombre/Descripción del evento". Below the search box, a red message "No hay resultados" is displayed.

- El enlace “Volver” de esta página debe volver a la página anterior para poder volver a realizar una nueva búsqueda (**BuscarEventos.jsp**).

A continuación, vamos a ver los pasos a seguir para construir nuestra aplicación.

1. **Crear el nuevo proyecto web “JavaEEGestionEventosEmisora”** con el **descriptor de despliegue de la aplicación**, es decir, con en el archivo **web.xml**, y configurarlo para que nuestra aplicación arranque en este caso con la página de inicio **Login.jsp**.
2. **Incluir las librerías JSTL** que vamos a usar en nuestras JSP's.
3. **Incluir el driver y crear un pool de conexiones**, en este caso **a la nueva BBDD “festivaldb2”**.

*NOTA: Debes crear primero la BBDD con el script proporcionado junto a este documento.*

4. En esta aplicación vamos a trabajar con eventos y usuarios, así que necesitas **crear el paquete `es.accenture.emisora.entidades` y crear las clases `Evento` y `Usuario`**.

*NOTA: Debes fijarte en las tablas de la BBDD para saber que atributos deben tener estas clases.*

5. Como vamos a lanzar nuestras propias Excepciones, debemos **crear el paquete `es.accenture.excepciones` y crear la clase `ExcepcionPropia`**. Dentro de esta clase creamos las constantes necesarias con los distintos mensajes de error y el constructor de parámetros con un parámetro de tipo String para el mensaje de la excepción.
6. Como vamos a seguir trabajando con el patrón MVC, necesitamos **crear el paquete `es.accenture.modelos` y crear las clases `modelo ModeloEvento` y `ModeloUsuario`** que se encarguen de conectar con la BBDD y realizar las diferentes operaciones.

Dentro de la **clase `ModeloEvento`** tendremos los siguientes métodos (operaciones contra la BBDD):

- **Método `getEventos`:** Este método se encargaba de devolver una lista con todos los eventos de la BBDD.
- **Método `getEventos(String criterio)`:** Este método se encargaba de devolver una lista con todos los eventos de la BBDD que cumplan el criterio de búsqueda, es decir, retorna los eventos que contengan el valor introducido como criterio en la búsqueda tanto en su nombre o descripción.

Dentro de la **clase `ModeloUsuario`** tendremos los siguientes métodos (operaciones contra la BBDD):

- **Método `getUsuario`:** Este método se encarga de buscar la información de un usuario en la BBDD, y dicha búsqueda la realizará por el usuario y la contraseña introducida.

*NOTA: Acuérdate, que ambas clases necesitan una variable o atributo de tipo `DataSource`, encargada de almacenar el pool de conexiones y el constructor de parámetros.*

Una vez que ya tenemos creadas la clase Modelo del patrón MVC, vamos a **crear el paquete `es.accenture.servlets` y crear el Servlet `ServletEventos`**. Este Servlet recibirá diferentes peticiones, y en función de la petición recibida deberá delegar dicha petición al controlador correspondiente, el cual se encargará de retornarnos la visa correspondiente para que el servlet se encargue de redirigirnos a ella.

<b>Acción/Petición</b>	<b>Controlador</b>
<b>"login"</b> → acción lanzada desde el botón de Login de la página Login.jsp	<b>ControladorIniciarSesion</b>
<b>"logout"</b> → acción lanzada desde el enlace "Cerrar sesión" de la página BuscarEventos.jsp	<b>ControladorCerrarSesion</b>
<b>"buscar"</b> → acción lanzada desde el botón de Buscar de la página BuscarEventos.jsp	<b>ControladorBuscarEventos</b>
<b>"vovler"</b> → acción lanzada desde el enlace "Volver" de la página MostrarEventos.jsp	<b>ControladorVolver</b>

NOTA: Acuérdate, que este Servlet necesita la referencia al pool de conexiones y crear la variable que almacene el pool de conexiones. Además, debes implementar el método doGet o doPost según corresponda o sea necesario.

8. Una vez creado el Servlet, podemos **crear el paquete es.accenture.controladores y crear la interfaz IControlador**, la cual nos sirve para predeterminedir un método común a todos los controladores que creemos, así que debemos declarar el siguiente método:

```
String procesarPeticion(HttpServletRequest request, HttpServletResponse response);
```

9. Una vez creada la interfaz, en el mismo paquete es.accenture.controladores podemos **crear los distintos controladores**:

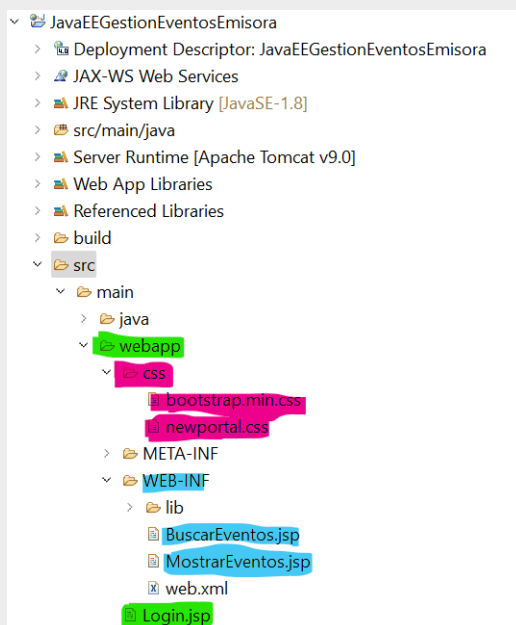
- **Controlador ControladorIniciarSesion**, dicho controlador se encarga de buscar el usuario y la contraseña introducida en la BBDD para recuperar la información del usuario (usar método getUsuario). Pueden pasar varias cosas:
  - Si los campos están vacíos, nos quedamos en la vista Login.jsp y mostramos el mensaje "Usuario/Password son obligatorios".
  - Si los campos están informados pero no existen en la BBDD, nos quedamos en la vista Login.jsp y mostramos el mensaje "Usuario/Password incorrecta".
  - Si los campos están informados y además existen en la BBDD, guardamos la información del usuario en sesión y accedemos a la vista BuscarEventos.jsp.
- **Controlador ControladorCerrarSesion**, dicho controlador se encarga cerrar la sesión y volver a la vista de Login.jsp.
- **Controlador ControladorBuscarEventos**, dicho controlador se encarga de buscar en la BBDD aquellos eventos que cumplan los criterios de búsqueda (usar método getEventos), es decir, pueden ocurrir varias cosas:
  - Si el campo de búsqueda está vacío, recuperamos todos los eventos de la BBDD y accedemos a la vista MostrarEventos.jsp con la información recuperada.



- Si el campo de búsqueda está informado, recuperamos aquellos eventos que contengan el valor introducido en su nombre o descripción y accedemos a la vista `MostrarEventos.jsp` con la información recuperada.
- Si el campo de búsqueda está informado pero no existen eventos en la BBDD que contenga dicho valor en su nombre o descripción, en ese caso nos quedamos en la vista `BuscarEventos.jsp` y mostramos el mensaje “No hay resultados”.
- **Controlador ControladorVolver**, dicho controlador se encarga simplemente de volver a la vista `BuscarEventos.jsp`.

10. **Incluir las vistas de la aplicación (ficheros JSP's) y hojas de estilo** que usan nuestras JSP's.

NOTA: Debes **copiar los distintos jsp's** proporcionados junto a este documento, de tal forma, que el fichero **Login.jsp** quede en la carpeta `webapp` del proyecto y los ficheros **BuscarEventos.jsp** y **MostrarEventos.jsp** queden en la carpeta `WEB-INF` del proyecto. Además, debes **copiar la carpeta css** proporcionada junto a este documento, con las hojas estilos dentro de la carpeta `webapp` del proyecto.



Una vez copiados, debemos realizar las modificaciones oportunas para que lancen las peticiones correspondientes y se comporte como se indicaba al comienzo del documento.

11. Una vez que tenemos todos nuestros componentes creados y modificados, ya podemos **probar aplicación**. Para probar la aplicación, simplemente debemos ejecutar el proyecto con el servidor, es decir, sobre el nombre del proyecto con el botón derecho damos a **Run As > Run a Server**.

**Asegúrate de que la aplicación funciona como se indica al principio del documento y que los datos que aparecen realmente son los que tenemos en la BBDD (puedes usar la herramienta de Workbench).**



# Gracias