



## **PRÁCTICA 3: DESARROLLO DE UNA APLICACIÓN WEB**

---

En esta práctica recopilamos todos los conocimientos aprendidos, en las sesiones tanto de teoría como de práctica, en referencia al desarrollo de *back-end* y *front-end* en web y buenas prácticas. Trabajaremos con tecnologías Java y J2EE contra base de datos MySQL (alojada en la UCO).

Tenga en cuenta los siguientes **aspectos normativos** referentes a esta práctica:

- Se entregará un **informe de prácticas** en formato PDF (máximo de 6 páginas) que deberá explicar y fundamentar las decisiones de diseño e implementación que ha adoptado el equipo, así como la relación de fuentes consultadas y un análisis de las dificultades encontradas.
  - Indique en el informe el nombre de los integrantes del equipo, así como la pila de tecnología utilizada para el desarrollo.
  - En el informe debe incluir el mapa de la aplicación desarrollada y las funcionalidades que ofrece, con clara referencia al MVC que cumple con cada funcionalidad.
  - En el informe se debe indicar el esquema relacional de la base de datos, así como los DAO de la capa de datos con acceso a estas tablas, y sus métodos.
  - El informe debe indicar las instrucciones de acceso, usuarios ya creados para prueba, y cualquier otra información relevante para la operación.
  - Se debe indicar en el informe cualquier punto adicional que haya incluido el equipo aparte de lo que se hubiera solicitado estrictamente en este enunciado.
- Además del informe, cada miembro del equipo entregará en privado el **informe de auto-/heteroevaluación** dispuesto en Moodle para tal fin. Sin este informe no será evaluado de esta práctica.
- Se hará entrega del **código fuente**, que debe estar completamente documentado, tanto las clases Java como los JSP y código de *front-end*, según proceda. Este código, al igual que cualquier otro artefacto asociado, debe estar claramente estructurado y organizado como se ha descrito en las sucesivas prácticas.
  - Se debe tener especial cuidado en el nombrado de paquetes y clases.
  - Los datos contenidos en la base de datos deben ser realistas.
  - Obsérvese que la aplicación tiene una parte significativa de margen de decisión por parte del equipo. Dado que se espera que estas decisiones sean

distintas por cada equipo de prácticas, deberán estar justificadas pertinentemente en el informe de prácticas.

- Se hará entrega del **fichero WAR** de la aplicación, que deberá ejecutar correctamente (incluyendo sus respectivas dependencias) en el Tomcat *standalone* (versión 8.5) dispuesto en la UCO.
- Se debe realizar un **vídeo explicativo** de no más de 10 minutos de duración, siguiendo el siguiente guion:
  - Una explicación de la estructura MVC, detallando su correspondencia con las funciones de la aplicación. Para este apartado, el equipo puede realizar la explicación mientras muestra el contenido del proyecto desde el IDE Eclipse, mostrando la jerarquía de directorios y el contenido de los ficheros más relevantes (flujo entre controladores y vistas, gestión de errores, acceso a base de datos, etc.). Este apartado no debe durar más de 5 minutos.
  - Una demostración realista del uso de la aplicación, que haga un recorrido completo como usuario espectador: registro, acceso, búsqueda de espectáculos por categoría, creación de una crítica y valoración de una crítica de otro usuario. Este apartado no debe durar más de 5 minutos.

Respecto a la entrega, considere los siguientes aspectos:

- Sólo hará entrega de la práctica un miembro del equipo a través de Moodle. Deberá subir a Moodle esta práctica la misma persona que entregase la primera práctica.
- Se hará entrega a través de la tarea Moodle dispuesta para ello de un archivo ZIP que encapsule el proyecto Eclipse (preferiblemente) o ficheros fuente de la práctica.
- Tanto el archivo PDF correspondiente al informe de prácticas como el vídeo explicativo deben estar incluido en la raíz del directorio del proyecto.
- Los informes de auto-/heteroevaluación son privados, por lo que **NO** deben incluirse en la entrega de prácticas, sino en una tarea específica para tal fin.
- El archivo ZIP que se entrega debe ser nombrado de la siguiente forma: GM<gm>\_<login>.zip donde <gm> es el número de grupo mediano y <login> es el nombre de usuario de la UCO de la persona del equipo que hace la entrega de esta y el resto de prácticas.

El incumplimiento de las normas de entrega (contenido, nombrado, etc.) implicará la consideración de práctica no entregada. La fecha de entrega se dispondrá en la tarea Moodle correspondiente. Igualmente, el retraso en la entrega supondrá la no calificación en la práctica.

**IMPORTANTE:** No se admitirá entrega alguna fuera de plazo o a través de un medio distinto a la tarea de Moodle (p.ej. envío por email).

## Funcionalidad de la aplicación web

Se desea implementar una aplicación web para la compartición de críticas de espectáculos, que permita además consultar la agenda de espectáculos disponibles en la ciudad. La aplicación debe permitir el registro de usuarios (espectadores), así como la modificación de sus datos personales y la opción de darse de baja de la aplicación. Además, la aplicación debe contemplar estas mismas opciones para usuarios administradores. Según el tipo de usuario, la aplicación dará acceso a unas funcionalidades u otras, debiendo comprobar que el usuario que accede a una determinada funcionalidad tiene permiso para ello implementando un mecanismo de control de acceso (*Access Control List*, ACL).

A los usuarios administradores les está permitido realizar las siguientes operaciones:

1. Ver el listado de usuarios (espectadores o administradores) registrados en la aplicación, incluyendo su fecha de registro y último acceso.
2. Dar de alta un espectáculo de alguno de los tipos definidos en la práctica 1:
  - a. Espectáculo puntual, es aquel que solo tiene una fecha y hora de representación.
  - b. Espectáculo de pase múltiple, que consta de varias fechas de representación.
  - c. Espectáculo de temporada, que se celebra periódicamente un día de la semana entre unas fechas de inicio y fin.
3. Añadir, cancelar o modificar sesiones para los distintos tipos de espectáculos, según sus restricciones en cuanto a número de sesiones y fechas disponibles.

A los usuarios espectadores les está permitido realizar las siguientes operaciones:

1. Buscar espectáculos por título o categoría.
2. Publicar una crítica para un espectáculo que ya se ha celebrado (al menos una sesión).
3. Consultar las críticas de un espectáculo, una vez seleccionado su título.
4. Eliminar una crítica de un espectáculo por parte del usuario que la creó.
5. Valorar la utilidad de una crítica publicada por otro usuario.

Todos los usuarios deberán ser identificados según su correo electrónico, y acceder al sistema mediante una contraseña. Una vez accedido, el usuario observará un menú de opciones adaptado según si se trata de un usuario administrador o espectador.

**Opcionalmente**, se podrán implementar otras funcionalidades con el objetivo de mejorar la calificación final de prácticas. Estas funcionalidades solo se tendrán en consideración si la práctica está aprobada (nota superior a 5), y tendrán un valor máximo de 2 puntos.

#### *Operaciones adicionales del usuario administrador:*

1. Cancelar un espectáculo, lo cual implica la cancelación de todas sus sesiones programadas.
2. Registrar la venta de localidades para un espectáculo y sesión determinados.

#### *Operaciones adicionales del usuario espectador:*

1. Consultar el número de localidades disponibles para un espectáculo, dada una fecha de representación.
2. Buscar próximos espectáculos con entradas disponibles, indicando o no una categoría específica.

### **Importantes consideraciones sobre el diseño y la implementación**

En cuanto a diseño, el patrón arquitectónico MVC debe ser la columna vertebral del desarrollo. En su mayoría, los controladores se implementarán como *servlets*, salvo que las condiciones de seguridad y privacidad permitan el uso de un JSP. Se hará uso de los ficheros de propiedades conforme a lo indicado en el ejercicio 2 de la Práctica 2. El fichero *web.xml* deberá configurarse adecuadamente para su uso con *servlets* y conforme a los parámetros de contexto. Igualmente, se deberá llevar a cabo control de errores y excepciones, tanto a nivel de JSP/*servlets* como de código de Java.

En el *front-end*, se empleará HTML5 y CSS3 para el desarrollo de la interfaz de usuario. *Javascript* se utilizará para la validación de formularios y paginación de resultados de espectáculos y críticas. También se valorará positivamente el uso de este lenguaje para aligerar en la medida de lo posible las llamadas al servidor (por ejemplo, en el filtrado). En el informe de prácticas es recomendable indicar qué funcionalidades de *Javascript* se han desarrollado con este fin.

Se exige de la realización de ciertas características que se exigirían a una aplicación con estas funcionalidades: autenticación cifrada, por tokens o por terceros, *front-end* responsivo, llamadas asíncronas a servidor, etc. Si algún equipo desea implementar alguna de estas características como extra, deberá hacerlo bajo su propia responsabilidad, de forma autónoma y siempre indicándolo adecuadamente en el informe de la práctica.

Es importante que se tenga en cuenta que se observará y evaluará con especial atención la calidad de la implementación, acogiéndose a los criterios de validación y buenas prácticas en el desarrollo que se han estudiado a lo largo del curso. En la medida de lo estudiado, se espera un resultado profesional y acogido -al menos- al enunciado de esta práctica. En caso de no cubrir el alcance completo esperado en la práctica, se deberá indicar expresamente tanto en el informe de prácticas como en el informe de auto-/hetero-evaluación.