

# Aplicaciones Distribuidas (AD)

## Práctica 2: Desarrollo de una aplicación web

En esta práctica se va a desarrollar una aplicación web con *Netbeans*. La práctica durará 3 sesiones y deberá realizarse una única entrega. Hay que entregar el código desarrollado junto con un informe del trabajo realizado.

En este informe se tiene que explicar el funcionamiento de la práctica, incluyendo funcionalidades extra añadidas y posibles modificaciones a la estructura de la aplicación presentada en este enunciado. Se tiene que entregar todo en un único archivo comprimido, que no debe contener los ficheros .jar que utiliza la aplicación web.

Revisad la rúbrica de evaluación para saber qué aspectos se van a evaluar en esta práctica y su peso sobre la nota de la misma. Consultad la fecha de entrega en el Racó.

### Desarrollo de la práctica

La práctica consiste en desarrollar una aplicación web para gestionar imágenes JPEG por parte de los usuarios registrados. Con esta aplicación los usuarios podrán hacer login, registrar nuevas imágenes, modificar los registros de las imágenes que hayan creado, listar las imágenes que hay en el sistema y buscar imágenes a partir de sus características. La información se debe almacenar en una base de datos. Podéis utilizar las de la práctica 1 como referencia. Sólo es necesario utilizar una de las dos, *sqlite* o *JavaDB*.

Es obligatorio implementar sesiones para poder acceder a las funcionalidades del sistema. En caso de que un usuario intente acceder sin haber hecho login, será redirigido a la página inicial, *login.jsp*. Es necesario que todas las páginas sean de tipo *jsp* para poder comprobar si el usuario tiene una sesión abierta o no.

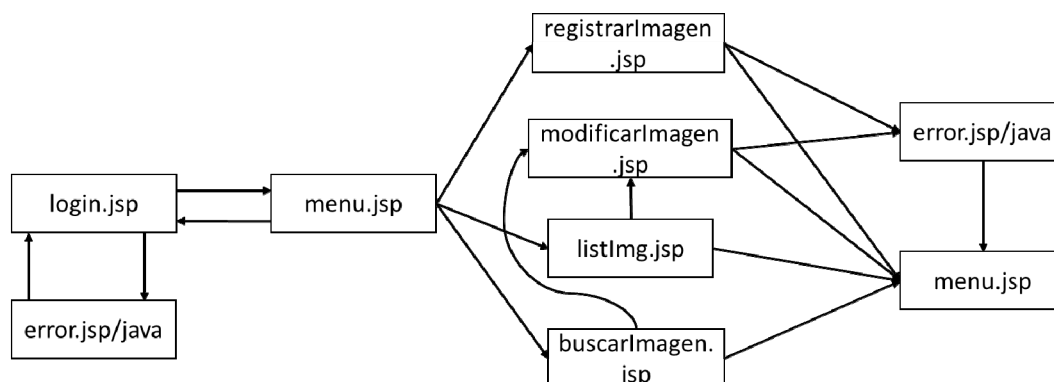
Se deben implementar (como mínimo) los siguientes componentes (páginas *jsp* y *servlets*):

- ***login.jsp***: Página que contiene el formulario que pide los datos al usuario para entrar en el sistema. Estos datos son usuario y password.
- ***login.java***: Servlet que recoge los datos de la página de login, comprueba que el usuario esté en la base de datos. Si está, nos redirecciona al menú. Si no está o el password es erróneo nos redirecciona a la página de error.
- ***menu.jsp***: Página que muestra el menú de opciones cuando el usuario introduce correctamente sus datos. Esta página sólo contiene enlaces a las páginas *jsp* de gestión de imágenes que se detallan a continuación.
- ***registrarImagen.jsp***: Página que pide los datos de una imagen para darla de alta en la aplicación. Los datos de una imagen deben ser (como mínimo), identificador de la imagen (no se le pide al usuario, se asigna automáticamente), título, descripción, palabras clave, autor, fecha de creación, fecha de alta en el sistema y nombre del

fichero que la contiene. Los datos de la imagen que no se generen automáticamente se tienen que pedir al usuario con un formulario. La imagen también se tiene que subir al sistema (campo de formulario HTML tipo file) y se almacenará en un directorio fijo de la aplicación web.

- **registrarImagen.java**: Servlet que recoge los datos de la imagen y los guarda en la base de datos. También debe leer la imagen y guardarla en un directorio de la aplicación web. En caso de que se pueda registrar la imagen correctamente, mostrará un mensaje y dará la opción de volver al menú o registrar otra imagen. En caso de error, redireccionará al usuario a la página de error.
- **modificarImagen.jsp**: Página que permite modificar los datos de una imagen registrada por un usuario en el sistema. A esta página se puede llegar desde las páginas de listado o búsqueda, que se explican a continuación.
- **modificarImagen.java**: Servlet que recoge los datos de la imagen a modificar y actualiza la base de datos y/o el fichero de la imagen. En caso de que se pueda modificar la imagen correctamente, mostrará un mensaje y dará la opción de volver al menú. En caso de error, redireccionará al usuario a la página de error.
- **list.jsp**: Página que lista todas las imágenes registradas en el sistema. Las imágenes deben tener un enlace que permita visualizarlas. Esta funcionalidad sólo la pueden realizar los usuarios registrados en el sistema. Para las imágenes que ha registrado el usuario tiene que aparecer la posibilidad de modificar la imagen (enlace a la página modificarImagen).
- **buscarImagen.jsp**: Página que permite buscar imágenes a partir de uno o varios campos asociados a la imagen. Para las imágenes que sean del usuario que hace la búsqueda tiene que aparecer la posibilidad de modificar la imagen (enlace a la página modificarImagen).
- **buscarImagen.java**: Servlet que recoge los datos de búsqueda y devuelve el resultado de la búsqueda en forma de listado. Si la búsqueda no devuelve ningún resultado se debe informar al usuario, dándole la posibilidad de realizar otra búsqueda o de volver al menú.
- **error.jsp/java**: Página que indica que ha habido un error y muestra un enlace para volver a la página de login o al menú, dependiendo de la página de la que vengamos.

El flujo básico entre páginas se muestra en la Figura 1 (podéis añadir más páginas y/o conexiones entre páginas si lo creéis necesario):



**Figura 1.** Flujo entre páginas

**Notas:**

- En la Figura 1 no aparecen los servlets que recogen los datos de los formularios que hay en las páginas jsp. Por ejemplo, para ir de login.jsp a menu.jsp es necesario pasar por un servlet que compruebe que el usuario existe en la base de datos.
- Se pueden añadir más funcionalidades a la aplicación como por ejemplo el registro de usuarios, el borrado de imágenes del sistema o el logout. Se puede cambiar el servlet error.java por una jsp.
- Se pueden incluir hojas de estilo para modificar el aspecto de las páginas. El uso de hojas de estilo (CSS) tendrá un impacto pequeño en la nota de la práctica, pero si la práctica no funciona, por usar hojas de estilo no se va a aprobar esta entrega.
- Todas las modificaciones realizadas se deben explicar en el informe de la práctica.

**Algunas referencias relacionadas con subida de ficheros en Aplicaciones Web J2EE**

<https://docs.oracle.com/javaee/6/tutorial/doc/glraq.html>

<https://www.erpgreat.com/java/upload-files-using-jsp-and-servlet.htm>

<https://docs.oracle.com/javaee/7/tutorial/servlets016.htm>