

Practica 3

Índice

Introducción	3
Server	3
WS.java.....	3
Image.....	4
callsSQL.....	4
Client_Application	4
Main	4
callsSQL2.....	5

Introducción

En esta práctica hemos tenido que desarrollar una aplicación web y una aplicación con un servidor externo a los proyectos, por tanto, la base de datos por ejemplo no se ha podido acceder directamente, sino que se ha hecho a través del server.

Server

El servidor de esta práctica es el que se encarga de almacenar todas las imágenes que se registran tanto en la aplicación web como en la aplicación. Además, también se encarga de la gestión de la base de datos, es el que hace todas las consultas y accesos a ella. Por consiguiente cuando una de las dos aplicaciones necesita alguna cosa de la base de datos ha de acceder primero al servidor para conseguirla.

Hemos creado tres paquetes para diferenciar las distintas clases. Un paquete lo hemos llamado basedatos y allí hemos añadido nuestra clase callsSQL.java. Otro paquete que hemos añadido es el de modelo donde está la clase Image.java. El último paquete creado es el de servicio y en el hemos metido nuestra clase WS.java

WS.java

Esta clase contiene todas las operaciones necesarias que se vamos a llamar de forma remota. Es la clase que se conecta con nuestro DAO para acceder a la base de datos.

Todas las operaciones que queramos hacer desde nuestra aplicación o desde nuestra aplicación web, en caso de querer acceder a la base de datos para hacer alguna acción con una imagen, deberán pasar primero por el WS para que este se comuniquen con el DAO y este con la base de datos. Esto está hecho así por un tema de consistencia, pues el objetivo es emular que nuestro servidor no está en la misma máquina que en la que estamos trabajando nosotros.

Image

Esta clase contiene todos los atributos necesarios para definir una imagen en la práctica.

El motivo de la existencia de esta clase es que podamos pasar las imágenes y todo su contenido de forma correcta entre el servidor y la aplicación o la aplicación web.

callsSQL

Esta clase emula nuestro DAO, accede a través de llamadas de funciones a la base de datos para las diversas acciones que se pueden realizar con una imagen como registrarla, eliminarla o modificarla.

Client_Application

Otro apartado de la práctica que teníamos que desarrollar es una aplicación cliente. En esta aplicación hemos creado dos paquetes distintos, uno es donde hemos guardado nuestra clase principal, *main*, y la otra es el paquete basedatos donde hemos guardado la clase *callsSQL2.java*

Esta aplicación funciona desde consola y se pueden probar las distintas operaciones que hacer. La aplicación da una serie de opciones que realizar mediante un listado numérico. Una vez elegida la operación, redirige a la función específica donde se describe paso a paso a través del terminal que debe hacer el usuario.

Hemos creado un servicio web de cliente que se conecta al servidor que hemos creado. Por esta razón, todas las operaciones que se realizan con una imagen se mandan al servidor y desde allí se opera. Para el registro de usuarios en cambio hemos creado una base de datos local que comprueba si el usuario existe o no.

Main

Es la clase principal donde se hacen todas las operaciones.

En esta clase tenemos las funciones necesarias para comprobar que la aplicación funciona. La función *main* es la primera que se ejecuta cuando se arranca la aplicación. Esta es la que muestra

a través de pantalla el listado con las distintas operaciones disponibles. Una vez seleccionada una de las posibles acciones, se redirige a cada una de las funciones correspondientes.

Al ser una aplicación que funciona desde terminal, se notifica al usuario que haga acciones concretas para que pueda funcionar todo de manera correcta. Un ejemplo de ello es en la función que usamos para registrar una imagen, se solicita al usuario que inserte en el terminal el path completo de la imagen para poder registrarla de manera correcta.

callsSQL2

Esta clase se encuentra en el paquete de basedatos y se encarga de comprobar los usuarios que usan la aplicación. En caso de que intente acceder un usuario que no pertenece aplicación, se muestra un mensaje informativo y se termina la aplicación.

Servlet-Application

La última aplicación para desarrollar de la práctica ha sido una aplicación web como la de la práctica 2 con la diferencia que ahora no tenemos el servidor en la misma aplicación.

La mayoría de las clases JSP se han quedado iguales pues solamente mostraban lo que había en el servlet. Las clases java en cambio si que se han tenido que modificar ya que estas son las encargadas de gestionar las acciones con las imágenes.

Como ya hemos dicho, nuestro servicio no está alojado en la aplicación web sino que esta en otra parte. Para gestionar esto, hemos creado un servicio web de cliente que se conecta al servidor. Por lo tanto, igual que en la aplicación del cliente, nuestras operaciones relacionadas con las imágenes no se harán de forma local sino que se harán de forma remota.

Para la gestión de usuarios en cambio, hemos creado un DAO local que se conecte a la base de datos para comprobar si los usuarios que quieren acceder a la aplicación existen o no.

En esta aplicación web hemos creado dos paquetes, uno llamado basedatos donde hemos guardado nuestro DAO y otro paquete llamado servicio donde hemos guardado todas nuestras clases java.