

Informe de la Pràctica 2

Pere Joan Martorell Calaf (46357187)

Bernat Farrero Badal (47870680)

1. Introducció

En aquesta pràctica hem dissenyat un servlet en Java amb les mateixes funcionalitats que el cgi de la pràctica anterior atenent els lloguers de cotxes amb el mètode *doGet()* i llistant-nos amb el mètode *doPost()*.

La principal diferència entre els servlets i els cgi's és que quan fem servir servlets optem per tenir una sola instància d'execució per totes les peticions que es fan al servidor, contràriament al que passa amb els cgi's, on per cada petició s'executa un procés nou. També s'ha de tenir en compte que en els servlets hi ha persistència de la informació entre peticions mentre que en els cgi's hem de guardar les dades en un fitxer per tal que no es perdin entre petició i petició.

Per altra banda, amb els servlets hem de vigilar amb els accessos concurrents a variables o parts del codi crítiques. Aquest problema es pot resoldre amb l'ús de monitors o semàfors.

2. Entorn

Per poder utilitzar servlets hem instal·lat Tomcat, un servidor web escrit en Java que té suport per a servlets. També hem definit la variable d'entorn *\$classpath* de manera que Java pugui trobar les classes dels servlets.

Per fer funcionar el nostre servlet cal que col·loquem el codi compilat dins la carpeta *classes* que hi ha dins del directori de la nostra *web-app*. També hem de modificar el fitxer *web.xml* per tal que el Tomcat sàpiga que existeix el nostre servlet indicant-li el nom del servlet i la seva *url*. Sempre que modifiquem aquest fitxer caldrà reiniciar el servidor.

3. Implementació

3.1 Guardar els lloguers

Per emmagatzemar els lloguers hem creat una llista d'una classe auxiliar anomenada *dadesLloguer* on hi guardem la informació referida al model del cotxe, tipus de motor, import, etc.

Per poder capturar i tractar els paràmetres que arriben per mètode GET hem de reescriure i sobrecarregar la funció *doGet(..)*. Dins d'aquesta rutina obtenim els valors dels paràmetres que ens arriben del formulari de lloguers i

comprovem que siguin vàlids. Fem la conversió d'aquests paràmetres i encapsulem tot això dins un *try-catch* per evitar les possibles excepcions derivades de valors nuls o no coherents amb el tipus al qual volem fer la conversió.

Seguidament calculem l'import total del lloguer i guardem tots els paràmetres dintre la llista de lloguers que hem creat inicialment. Per evitar que les peticions al servidor accedeixin concurrentment a la mateixa llista de lloguers fem servir un semàfor. Això fa que el codi contingut dintre el *synchronized* s'executi com un sol bloc.

Per acabar, escrivim totes les dades del lloguer pel canal d'escriptura, que serà la resposta que rebrà el navegador del client.

3.2 Llistar els lloguers

Llistem els lloguers mitjançant la sobrecàrrega del mètode *doPost()*.

Primer de tot obrim un canal de sortida i establim com a *text/html* el contingut MIME de la resposta. Obtenim el nom d'usuari i la contrasenya i comprovem si efectivament es corresponen amb els autèntics. De ser així, recorrem la llista de lloguers amb un iterador i escrivim tots els lloguers pel canal de sortida.