



Escola Universitària
Politécnica de Mataró

Enginyeria Tècnica en Informàtica de Gestió

Web Paisatges del Món

**NOM: SERGIO LUNA DÍAZ
JOAN JOU i MAJÓ**

TARDOR 2009

A la meva família i la meva parella per la seva comprensió i ajuda.

Agraïments al professor Joan Jou i Majó, per la seva ajuda i direcció en aquest projecte.

Sumari

Aquest projecte s'ha plantejat per a que un usuari pugui compartir fotografies i vídeos propis fets en un viatge, donar una descripció i situació de la imatge perquè els altres usuaris puguin fer-se una idea del paisatge. Per a realitzar la gestió del posicionament de la fotografia o vídeo s'ha utilitzat el servei de Google Maps, per guardar les dades s'ha utilitzat la base de dades MySQL i per guardar les fotografies i vídeos s'ha simulat un FTP en el sistema.

Sumario

Este proyecto se ha planteado para que un usuario pueda compartir fotografías y vídeos propios hechos en un viaje, dar una descripción y situación de la imagen que los demás usuarios puedan hacerse una idea del paisaje. Para realizar la gestión de posicionamiento de la fotografía o vídeo se ha utilizado el servicio de Google Maps, para guardar los datos se ha utilizado la base de datos MySQL y para guardar las fotografías y videos se ha simulado un FTP en el sistema.

Abstract

This project has been raised for a user to share photographs and videos themselves made on a trip, give a description and location of the image that others can get an idea of the landscape. To make the management of positioning of the photograph or video have used the Google Maps service, to save the data we used the MySQL database and to store photos and films has simulated an FTP in the system.

Índex

1. Introducció	3
1.1. Motivació del projecte	3
1.2. Objectius	4
1.3. Requisits tecnològics	5
2. Estudi de mercat	7
2.1. La web de “Fotos de lugares con encanto”	7
2.2. La web de “Rutas del Sur.es”	8
2.3. Taula comparativa	10
3. Metodologia i tecnologia	13
3.1. Tecnologia i entorn de desenvolupament	13
3.2. Metodologia	15
3.3. Planificació i pressupost	16
3.4. Pressupost	18
4. Anàlisi requeriments	21
4.1. Estudi de requeriments	21
4.2. Casos d’ús	24
5. Disseny de l’aplicació	29
5.1. Disseny de la interfície gràfica	29
5.2. Domini del projecte	32
5.3. Disseny de la base de dades	33
6. Desenvolupament amb NetBeans i Framework Struts	35
6.1. Introducció a Struts	35

6.2. Què és Struts?	36
6.3. Com funciona els Struts?	36
6.4. Exemple línia d'execució cas d'ús Login usuari	41
6.5. Exemple línia d'execució cas d'ús Afegir fotografia	45
7. API Google Maps.....	53
8. FormFile i File.....	59
8.1. Afegir un arxiu a un FTP.....	59
9. Proves.....	65
10. Ampliacions i aplicacions del projecte.....	71
10.1. Ampliacions del projecte.....	71
10.2. Aplicacions del projecte	71
11. Conclusions	73
I Annex Connectar JAVA-MySQL	77
II Annex Workbench 5.2 OSS	81

1. Introducció

1.1. Motivació del projecte

La motivació per realitzar aquest projecte era la de realitzar una proposta de pàgina web en la qual persones que en el seu temps lliure han realitzat viatges, excursions... puguin compartir les seves fotografies, vídeos i vivències dels viatges realitzats. També es vol reflectir la possibilitat d'explicar les experiències i ser valorades per altres usuaris.

D'un bon principi es va plantejar la idea d'utilitzar el Google Maps¹ com a eina per desenvolupar la funcionalitat de creació i posicionar les fotografies i vídeos dels usuaris, ja que és una eina molt coneguda en l'àmbit d'Internet. És una eina gratuïta, fàcil d'utilitzar i amb característiques suficientment potents per dur a terme el projecte. Les seves principals característiques són la gran interacció que pot assumir un usuari amb els mapes, que estan relativament actualitzats, la possibilitat de canviar vistes en tot moment a satèl·lit, mapa o una combinació de les dues, obtenir o afegir informació concreta d'un element del mapa (amb l'ajuda de marcadors, globus de text,...), etc.

En els primers moments del projecte, es va considerar l'opció de pujar les fotografies i vídeos a un FTP (que en local seria un FTP simulat amb una carpeta dins del projecte) amb l'ajuda d'un programa de tercers com seria el FileZilla², però després de realitzar algunes consideracions d'instal·lacions, seguretat i integritat del sistema per part de l'usuari, s'ha optat perquè el propi sistema sigui l'encarregat de realitzar aquesta tasca.

L'aparença i l'estructura de la pàgina és un estil similar a un fòrum o bloc, ja que actualment són estructures que els usuaris estan molt més acostumats a veure i es sentiran molt més còmodes en la navegació.

1.2. Objectius

El projecte es podria dividir en tres clares parts. Serien els objectius que hauria de complir el projecte, per tal de ser viable i atractiu per a un usuari qualsevol.

La primera part seria que qualsevol persona que accedeixi a la pàgina web, sense cap tipus de bloqueig ni restriccions, però no tindria uns privilegis que si tenen els usuaris registrats. Les accions que podran realitzar l'usuari seran les següents:

- Veure qualsevol tipus de vídeo o fotografia. La seva situació en el Google Map i les seves característiques sense cap problema.
- Poder puntuar un vídeo o una fotografia amb la puntuació que ell consideri oportuna.
- Entrar en contacte amb l'administrador per qualsevol dubte o pregunta.
- Realitzar cerques de continguts en el site.

La segona part seria la dels usuaris registrats, han hagut d'emplenar un formulari previ, per tal que l'administrador tingui informació dels usuaris registrats en el sistema i poder tenir un control dels arxius pujats al FTP. Les accions que podran realitzar:

- Realitzar les operacions que fan els usuaris sense registrar.
- Poder pujar fotografies i vídeos al site per tal que altres usuaris puguin veure-les.
- Poder modificar els atributs de les seves fotos o vídeos segons cregui convenient l'usuari (respectant uns atributs per motius de seguretat i integritat del sistema).
- Esborrar les seves fotografies o vídeos que l'usuari cregui convenient.
- Consultar i modificar el seu perfil en tot moment.
- A part, si l'usuari no recorda el seu login podrà consultar-ho emplenant un formulari previ.

I la tercera part seria la d'administrador, que estaria destinada a dur una gestió d'usuaris, fotografies i vídeos. Aquesta part és d'accés restringit, per tant, només un usuari amb la credencial d'administrador hi pot accedir. Les accions que podrà realitzar seran les següents:

- Podrà realitzar les operacions que fan els usuaris no registrats.
- Control d'usuaris. Consultar, eliminar, modificar dades d'usuaris que cregui convenient.
- Control de fotografies i vídeos. Consultar, eliminar, modificar fotografies i vídeos pels motius que cregui convenient.

1.3. Requisits tecnològics

Es demana que el Host on estarà allotjat la pàgina web sigui un servidor capaç d'executar pàgines dinàmiques amb tecnologia JSP, Struts.

Referent a la base de dades el servidor haurà de tenir instal·lat i configurat Apache⁷ i MySql⁸, perquè el projecte està basat en aquesta tecnologia de base de dades però, realitzant unes petites modificacions (canviar la manera de connectar) es podria adaptar a totes les base de dades.

Per poder accedir a la pàgina web seria ideal que l'usuari disposés d'una connexió ADSL per millorar la transferència de les fotografies i vídeos, i reduir l'espera de la resposta entre peticions de l'usuari.

2. Estudi de mercat

A Internet hi ha un gran número de pàgines que tracten el tema de fotografia de paisatges, però la majoria són blocs o pàgines personals de gent anònima. La gran part només són pàgines de presentació de fotografies, i no van més enllà de mostrar la fotografia, cosa que pot avorrir els usuaris. Però en altres casos són pàgines que mostren les fotografies amb un segon objectiu, com per exemple, vendre viatges a aquests llocs.

S'ha escollit dues pàgines per analitzar, una d'un usuari anònim, *Imatge2.1.1* i l'altre de la comunitat autònoma d'Andalusia, *Imatge2.2.1*.

2.1. La web de “Fotos de lugares con encanto”

Aquesta web és molt simple, només té com a objectiu mostrar fotografies sense que hi hagi cap acció pàgina versus usuari.



Imatge 2.1.1: Homepage de <http://www.turismoyfotos.net>

En aquest site només té la idea de mostrar fotografies de paratges d'Espanya, dóna informació molt curta respecte el paisatge o quasi nul·la. No presenten cap referència a mostrar vídeos dels paisatges.

Respecte a la cartografia, si mostra la situació on s'ha realitzat la foto però, d'una manera molt rudimentària, amb un mapa del Google Earth però, la situació exacte, marcada amb el paint o un editor d'imatges.

No hi ha accions que l'usuari pugui realitzar en la pàgina. No hi ha un registre d'usuaris, per tant, un usuari no hi pot aportar fotografies a la pàgina directament. Tampoc un usuari no pot comentar ni valora una fotografia concreta. L'única possibilitat de participar en la pàgina és entrar en contacte amb l'administrador via e-amil.

2.2. La web de "Rutas del Sur.es"

Aquesta pàgina és molt més completa, és de la comunitat d'Andalusia.



Imatge 2.2.1: Homepage de <http://www.rutasdelsur.es>

Com podem observar, té com a objectiu donar a conèixer la comunitat amb més profunditat per a usuaris de tota Espanya, també mostra fotografies i rutes que es poden realitzar en tota la comunitat. Però encara falten opcions que el nostre projecte té com a objectiu reflectir.

Tot i ser una pàgina d'un organisme oficial, té mancances. Com per exemple, quan es visualitza una fotografia no mostra una descripció (història, situació, com arribar, ...), no presenten cap referència a mostrar vídeos dels paisatges, ni de rutes.

Respecte a la cartografia, no mostra en cap mapa la situació on s'ha realitzat la foto, ni cap manera de saber la situació exacta de la fotografia, només pel seu nom, el nom que ha posat l'usuari que l'ha pujat o bé el propi administrador. Les rutes no tenen un mapa que representi com realitzar la ruta, només hi ha una breu descripció realitzada per l'usuari que ha pujat la ruta o bé el propi administrador.

Les accions que pot realitzar un usuari en la pàgina és molt més rica. Hi ha un registre d'usuaris, per tant, els usuaris poden aportar imatges o "àlbums", rutes a la pàgina. Però l'únic inconvenient és que un usuari no pot valorar els àlbums d'altres usuaris, però sí comentar les rutes d'altres usuaris.

2.3. Taula comparativa

A continuació podem observar una taula resum de les característiques que tenen en comú i les diferències de les pàgines analitzades amb el projecte final de carrera (Web Paisatges del Món).

Característiques	Webs analitzades		
	Fotos de lugares con encanto	Rutas Del Sur.es	Web Paisatges del Món
Gestió administrativa dels usuaris	No	Si	Si
Gestió administrativa de fotografies	Si	Si	Si
Gestió administrativa de vídeos	No	No	Si
Registre usuaris	No	Si	Si
Interacció amb Google Maps	No	No	Si
Visualització fotografies	Si	Si	Si
Visualització vídeos	No	No	Si
Gestió i creació de rutes	No	Si	No
Gestió de fotografies	No	Si	Si
Gestió de vídeos	No	No	Si
Sistema de cerca fotografies i vídeos	Si	Si	Si
Sessió d'usuari	No	Si	Si
Interfície usable	Si	Si	SI

Taula 2.3.1: taula comparativa Webs

Després d'observar la *Taula 2.3.1* podem afirmar que la pàgina que té més punts en comú amb la pàgina del projecte (Web Paisatges del Món) és la de Rutas Del Sur.es.

S'observa que tenen quasi tots els punts en comú menys tres.

Un dels punts no comuns és el referent a vídeo, donat que la pàgina Rutas Del Sur.es no reflecteix l'opció de mostra vídeos de paisatges i el nostre projecte sí té com a objectiu fer-ho.

També no coincideixen la pàgina de Rutas Del Sur.es amb la del projecte, en el punt del Google Maps, ja que Rutas Del Sur.es no situa les fotografies de manera exacta en un mapa i el nostre projecte sí.

I per últim, la pàgina de Rutas Del Sur.es reflecteix la possibilitat de crear i gestionar rutes, però el nostre projecte no té aquesta característica com a objectiu.

3. Metodologia i tecnologia

3.1. Tecnologia i entorn de desenvolupament

El projecte s'ha desenvolupat en un sistema operatiu basat en Windows (WinNT). A continuació es mostra tot el software que s'ha utilitzat en el desenvolupament del projecte:

Sistema operatiu:

- Windows Vista i Windows 7.

Software addicional:

- Edició fotogràfica: Spesoft Image Converter.
- Edició de vídeo: Free Video Converter.
- Navegadors: FireFox 3.5.6 i Internet Explorer 8.

Software desenvolupament de l'aplicació web:

- Llenguatge Web Java: NetBeans IDE 6.7.1.
- MySQL Server 5.1.
- Gestor base de dades: MySQL Workbench 5.2 OSS.
- CSS i XHTML per la presentació de dades.
- Javascript / AJAX per la interacció amb l'usuari.
- API Google Maps.

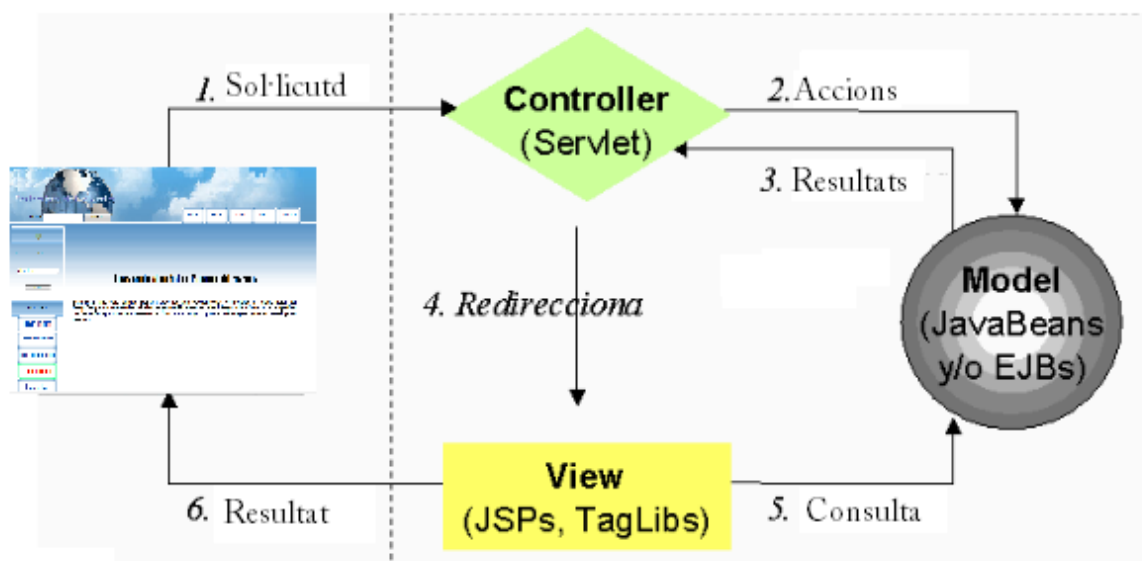
La tecnologia ha de estar relacionada amb el llenguatge web escollit per realitzar el projecte, per tant, el llenguatge escollit per realitzar tot el projecte es Java (JSP i el Framework Struts).

L'API de Google Maps utilitza la tecnologia CSS per mostrar els mapes, Javascript i AJAX perquè el programador pugui fer la interacció del mapa amb l'usuari.

Com a navegador web utilitzat per al desenvolupament ha sigut l'Internet Explorer 8, ja que és el propi del sistema operatiu Windows i també el més usat pels usuaris. Però també s'ha usat el FireFox 3.5.6, perquè cada vegada hi ha més usuaris que volen fer ús de programari lliure, i per tant, s'ha de comprovar el correcte funcionament en els dos navegadors.

S'ha aplicat el MVC (Model-Vista-Controlador) gràcies al Framework Struts. Aquest patró separa les dades de l'aplicació, la interfície d'usuari i la lògica del negoci en tres capes diferents. El patró MVC és un dels més utilitzats en el desenvolupament d'aplicacions web. La vista genera la pàgina HTML, el model correspon a la lògica de negoci i també inclou les sentències amb la base de dades (on s'ha fet un patró Façana) i el controlador és el responsable de rebre les accions de l'usuari des de la vista i de controlar el procés d'acció (on s'ha fet el patró paquets per tal de separar les accions dels "Actions" segons qui les realitza).

La següent *Imatge 3.2.1* mostra la funcionalitat del patró MVC realitzat amb el Framework Struts.



Imatge 3.2.1: MVC Struts

3.2. Metodologia

La metodologia que s'ha empleat en aquest projecte a sigut iterativa i incremental.

Atès que els projectes programari són llargs és comú dividir el treball en “mini-projectes” (les dimensions del projecte és el principal principi per utilitzar les iteracions). Cada “mini-projecte” és una iteració que resulta en un increment. Les iteracions es refereixen a passos en el flux de treball, i els increments a un creixement en el projecte. Per a ser més efectives les iteracions han de ser controlades, és a dir, han de ser seleccionades i portades a terme d'una forma planejada i en un temps determinat, de manera que cadascuna de les iteracions constituïx un “mini-project” programari.

S'ha dividit el projecte iteracions, cada una d'elles fa referència a un cas d'ús, cada iteració té un temps determinat d'inici i un temps de finalització i s'han de complir en la mesura del possible. Cada vegada que finalitzem una iteració, comprovem si s'han complit els objectius realitzant proves de funcionalitat i estabilitat.

En el cas que aquestes proves no tinguin èxit, s'ha de tornar a la iteració anterior i comprovar si es pot realitzar algun canvi per tal de millorar la situació o avançar en la falla de la prova. En el cas que les proves tinguin èxit, s'avança a la següent iteració.

En tot moment del desenvolupament d'alguna iteració, es pot aturar i revisar o afegir nous requeriments al sistema per tal de millorar el seu funcionament, estabilitat o simplement donar-hi més funcionalitats.

3.3. Planificació i pressupost

Per a la planificació del projecte s'ha utilitzat el programa Gantt Project⁵ per reflectir el temps dedicat a cada iteració. En la *Imatge 3.3.1* es pot observar el diagrama de Gantt obtingut.

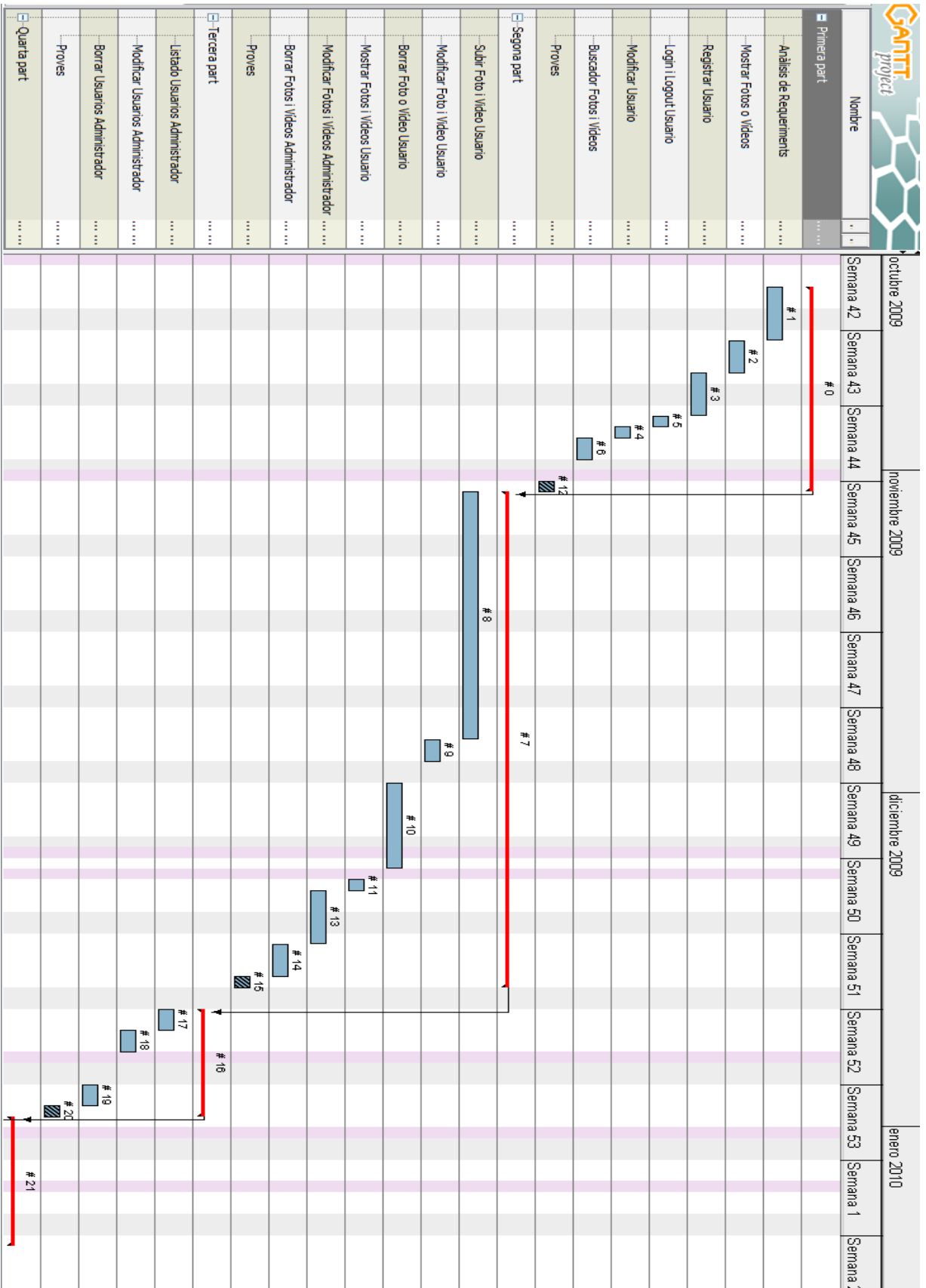
En la primera part de la planificació, es basarà en realitzar l'anàlisi de requeriments de l'aplicació web, crear l'accés dels usuaris a l'aplicació i la seva gestió. Aquesta gestió es basarà en habilitar l'accés a zones privades segons si són usuaris registrats o no, i també l'accés de l'administrador al seu propi menú privat. Al final de la iteració es comprovarà que s'hagi assolit amb èxit els objectius.

En la segona part, es desenvoluparà els casos d'ús principals que es consideren amb més complexitat. Per això, es reserva molt més temps per realitzar aquesta part de la planificació. Al final de la iteració es comprovarà que s'hagi assolit amb èxit els objectius.

En la tercera part, es desenvoluparà la part administrativa de la gestió d'usuaris, fotografies i vídeos. També es realitzarà casos d'ús secundaris que tenen importància per a un usuari no registrat, com per exemple, votar o denunciar una fotografia o vídeo. Al final de la iteració es comprovarà que s'hagi assolit amb èxit els objectius.

I finalment, en la quarta iteració, es procedirà a realitzar jocs de proves per comprovar el correcte funcionament de tota l'aplicació, i la documentació del projecte.

⁵ <http://www.ganttproject.biz/>



Imatge 3.3.1 Diagrama de Gantt

3.4. Pressupost

Per elaborar el pressupost d'aquest projecte, es considera que el temps màxim és de 300 hores (temps màxim fins la data d'entrega del projecte final de carrera).

Es va adquirir un equip informàtic sencer pel desenvolupament de l'aplicació web. Es va fer un estudi del software més adient pel desenvolupament del projecte (aquestes hores es consideren fora de les 300 hores), van ser un total de 50 hores a un cost de 20€/hora.

Les hores per realitzar el projecte són 300 hores 25€ hora i suposem que un enginyer informàtic té un sou entre 1200€ i 1500€ (informació obtinguda de www.infojobs.com).

COST DEL PROJECTE

Cost del treballador

1 persona en els 4 mesos de durada del projecte a 1350€ **5400€**

Cost de l'estudi

Estudi software previ al començament del projecte **1000€**

Cost informàtic (informació obtinguda ds.datasystem.com)

Kit PC HP CQ5204 E5300 **388,94€**

Monitor + perifèrics **118€**

Cost software

Windows 7 (gratuït proporcionat pel fabricant)	-
NetBeans 6.7.1	-
Adobe Dreamweaver CS4	239,99€
MySQL Server 5.1	-
MySQL Workbench 5.2. OSS	-
Navegadors (FireFox, Internet Explorer)	-
Eines ofimàtiques (Openoffice)	-

Cost del desenvolupament de l'aplicació web

25€hora per 300 hores (temps màxim)	7500€
TOTAL	14.647€

4. Anàlisi requeriments

4.1. Estudi de requeriments

L'estudi de requeriments es realitza segons els actors principals de l'aplicació. Ho dividirem en usuari públic, usuari registrat i administrador.

Els requeriments funcionals de l'usuari públic són:

- Visualitzar fotos o vídeos: L'aplicació web ha de permetre a qualsevol usuari que no s'hagi registrat en el sistema visualitzar fotografies i vídeos.

- Puntuar o denunciar fotos i vídeos: L'aplicació web ha de permetre a qualsevol usuari que no s'hagi registrat en el sistema puntuar o denunciar fotos i vídeos.

Els requeriments funcionals de l'usuari registrat són:

- Visualitzar fotografies o vídeos: L'aplicació mostrarà la imatge, la informació i situació referida a una fotografia o vídeo.

- Cercador de fotos i vídeos: L'aplicació web ha de permetre a l'usuari cercar fotografies o vídeos introduït el seu nom o bé el nom de la persona autora de la foto o vídeo.

- Recordar usuari: L'aplicació ha d'oferir la possibilitat que l'usuari recuperi la seva contrasenya.

- **Registre usuari:** L'usuari ha d'omplir un formulari, en el que es demana informació sobre ell, per poder ser membre de la web.

- **Sessió d'usuari:** L'usuari ha de poder iniciar sessió introduint un nom i una contrasenya. L'aplicació web no ha de permetre realitzar certes accions si no estan registrats, com afegir una foto o un vídeo. L'usuari s'ha de registrar a través d'un formulari.

- **Modificar usuari:** Per poder realitzar aquesta acció l'usuari ha d'haver iniciat sessió. L'usuari ha de poder modificar el seu perfil en tot moment.

- **Consultar perfil:** Per poder realitzar aquesta acció l'usuari ha d'haver iniciat sessió. L'usuari ha de poder consultar el seu perfil en tot moment.

- **Visualitzar fotos i vídeos d'un usuari:** Per poder realitzar aquesta acció l'usuari ha d'haver iniciat sessió. L'usuari ha de poder consultar el llistat de fotografies i vídeos que té en el sistema.

- **Afegir foto o vídeo:** Per poder realitzar aquesta acció l'usuari ha d'haver iniciat sessió. L'usuari ha de poder inserir les seves fotografies o els seus vídeos (complint unes normes establertes prèviament en el sistema), interaccionant amb el mapa del Google Maps i afegir informació referida a la fotografia o el vídeo. L'aplicació ha de ser capaç de guardar tant la informació com la situació de la fotografia o el vídeo per poder presentar-ho després.

- **Modificar foto o vídeo:** Per poder realitzar aquesta acció l'usuari ha d'haver iniciat sessió. L'usuari ha de poder modificar les seves fotografies o els seus vídeos (complint unes normes establertes prèviament en el sistema), interaccionant amb el mapa del Google Maps i modificar la informació referida a la fotografia o el vídeo. L'aplicació ha de ser capaç de guardar tant la informació modificada com la situació de la fotografia o el vídeo per poder presentar-ho després.
- **Eliminar foto o vídeo:** Per poder realitzar aquesta acció l'usuari ha d'haver iniciat sessió. L'usuari ha de poder eliminar les seves fotografies o els seus vídeos que no desitgi que continuïn en el sistema.

Els requeriments funcionals de l'administrador són:

- **Sessió d'usuari:** Per accedir a les accions d'administració, l'administrador ha d'iniciar sessió introduint un nom i una contrasenya.
- **Llistar usuaris:** Per poder realitzar aquesta acció l'administrador ha d'haver iniciat sessió. L'administrador podrà consultar tots els usuaris del sistema i les seves fotografies o vídeos.
- **Eliminar usuari:** Per poder realitzar aquesta acció l'administrador ha d'haver iniciat sessió. L'administrador podrà consultar tots els usuaris del sistema i eliminar l'usuari que incompleixi alguna normativa o si ho creu.

- **Modificar foto o vídeo d'un usuari:** Per poder realitzar aquesta acció l'administrador ha d'haver iniciat sessió. L'administrador ha de poder modificar les fotografies o els vídeos d'un usuari determinat. Podrà interaccionar amb el mapa del Google Maps i modificar la informació referida a la fotografia o el vídeo. L'aplicació ha de ser capaç de guardar tant la informació modificada com la situació de la fotografia o el vídeo per poder presentar-ho després.

- **Eliminar foto o vídeo d'un usuari:** Per poder realitzar aquesta acció l'administrador ha d'haver iniciat sessió. L'administrador ha de poder eliminar les fotografies o els vídeos que no desitgi que continuïn en el sistema, per incompliment d'alguna norma o bé per un altre motiu.

- **Gestionar foto o vídeos denunciats:** Per poder realitzar aquesta acció l'administrador ha d'haver iniciat sessió. L'administrador consultarà totes les fotos i vídeos denunciats i prendrà una decisió segons el seu parer d'eliminar la foto o vídeo, o bé cancel·lar la denuncia.

4.2. Casos d'ús

Els casos d'ús que s'exposen a continuació serà els més rellevants per el projecte:

Cas d'ús – Afegir Foto Usuari (el cas d'ús Afegir Vídeo Usuari és igual)

Actor principals: Usuari registrat

Pre-Condicions: L'usuari ha d'haver iniciat una sessió.

Post-Condicions: L'aplicació haurà guardat la informació de la foto i la seva situació en el Google Maps, i serà visible per a tots els usuaris.

Normal Flow:

1. L'usuari executa l'acció d'afegir foto.
2. El sistema mostra un mapa on l'usuari haurà de buscar la situació.
3. L'usuari busca la situació exacta de la foto en el mapa, fent clics fins trobar la situació.
4. El sistema recull les coordenades i guarda la informació.
5. L'usuari introdueix la informació referida a la foto: nom, títol, descripció, país, població, província, adreça, telèfon, orientació de la foto.
6. L'usuari fa clic en acceptar, el sistema recull les dades, les guarda i l'arxiu fotogràfic ho guarda en el FTP.

Alternate Flow

- 6.1 El sistema no pot guardar la fotografia, ha de mostrar un missatge d'error indicant el motiu de la falla.

Cas d'ús – Modificar Foto (el cas d'ús Modificar Vídeo és igual)

Actors principals: Usuari registrat o Administrador

Pre-Condicions: L'usuari ha d'haver iniciat una sessió.

Post-Condicions: L'aplicació haurà guardat la informació modificada de la foto i la seva situació en el Google Maps, i serà visible per a tots els usuaris.

Normal Flow:

1. L'usuari executa l'acció de modificar foto.
2. El sistema mostra un mapa on l'usuari haurà de buscar la situació.
3. L'usuari busca la situació exacta de la foto en el mapa, fent clics fins trobar la situació.

4. El sistema recull les coordenades i guarda la informació.
5. L'usuari introdueix la informació que vol modificar referida a la foto: descripció, població, província, adreça, telèfon, orientació de la foto. Per motius de seguretat i integritat del sistema algunes dades no es poden modificar.
6. L'usuari fa clic en acceptar i el sistema recull les dades noves i les guarda al sistema.

Alternate Flow

- 6.1 El sistema no ha pogut modificar les dades de la fotografia, ha de mostrar un missatge d'error indicant el motiu de la falla.

Cas d'ús – Eliminar Foto (el cas d'ús Eliminar Vídeo és igual)

Actors principals: Usuari registrat o administrador

Pre-Condicions: L'usuari ha d'haver iniciat una sessió.

Post-Condicions: L'aplicació eliminarà del sistema la fotografia escollida.

Normal Flow:

1. L'usuari executa l'acció d'eliminar foto.
2. El sistema mostra un llistat de les fotos que hi ha el sistema de l'usuari.
3. L'usuari busca la foto que desitja eliminar i fa clic sobre la foto.
4. El sistema identifica la foto i pregunta a l'usuari si vol eliminar la foto.
5. L'usuari fa clic a acceptar per eliminar la fotografia del sistema.
6. El sistema elimina les dades de la fotografia i elimina l'arxiu que està en el FTP de l'aplicació.

Alternate Flow

6.1 El sistema no pot eliminar la fotografia, ha de mostrar un missatge d'error indicant el motiu de la falla.

Cas d'ús – Visualitzar Foto (el cas d'ús Visualitzar Vídeo és igual)

Actors principals: Usuari.

Post-Condicions: L'aplicació mostrà la foto escollida per l'usuari.

Normal Flow:

1. L'usuari executa l'acció de visualitzar categoria foto.
2. El sistema mostra un llistat de fotos que compleixen la categoria escollida per l'usuari.
3. L'usuari busca la foto desitjada i fa clic sobre ella.
4. El sistema presenta la imatge i la seva informació a l'usuari.

Alternate Flow

2.1 El sistema no ha pogut mostrar les fotografies d'una categoria.

2.1.a El sistema no troba cap foto que estigui associada a aquesta categoria. Mostra un missatge d'error indicant el motiu de la falla.

2.1.b El sistema té un error i no pot mostrar el llistat de fotografies associades a una categoria. Mostra un missatge d'error indicant el motiu de la falla.

4.1 El sistema no ha pogut mostrar la fotografia desitjada, ha de mostrar un missatge d'error indicant el motiu de la falla.

5. Disseny de l'aplicació

5.1. Disseny de la interfície gràfica

L'aplicació consta de tres entorns diferenciats: el de l'usuari públic, el de l'usuari registrat i la part administrativa.

En la interfície es busca el màxim de simplicitat possible, per tal que l'usuari es mostri còmode en el seu ús. La interfície és comuna per als tres rols, l'únic que varia d'un rol a un altre és el menú privat per realitzar les operacions pròpies del rol.

Per tant, per tal de no complicar la navegació en l'aplicació s'ha optat perquè els menús fossin simples i directes, seran de tipus botó i llistat desplegable amb el nom a l'acció que es podrà realitzar si es fa clic en ell, *Imatge 5.1.1*, els menús sempre són visibles, això facilitarà la navegació de l'usuari. Les pantalles s'agrupen per temàtiques sense barrejar-les per tal que un usuari pugui treballar amb simplicitat.



Imatge 5.1.1: Pantalla inicial projecte

Per representar amb un icona el projecte es va considerar alguna manera de fer una idea global, el món,... però d'una manera atractiva, llavors es va escollir la següent imatge.



Imatge 5.1.2: Icona representatiu



Imatge 5.1.3: Logotip de la Web

S'ha escollit aquest disseny de logotip (*Imatge 5.1.3*) perquè després d'haver navegat i observat que la gran majoria de pàgines col·loquen el seu logotip està situat en la part superior esquerra utilitzant un 20% o 25% de la pantalla. Respecte a la navegació s'ha considerat crear dos menús. Un és el ja mencionat, un menú horitzontal desplegable que representarà les accions comunes per a tots els usuaris, situat en la part superior dreta. S'ha escollit aquest tipus de navegació perquè és molt més atractiva que el típic menú estàtic. I el segon és un menú vertical, situat a la part esquerra, sota el logotip. Aquest menú representa les accions pròpies de cada rol d'usuari (usuari públic, usuari registrat, administrador). En la *Imatge 5.1.4*, es pot observar les dues interfícies segons els rols (usuari registrat i administrador, perquè el rol d'usuari públic és el de la *Imatge 5.1.1*).

Interfície usuari registrat



Interfície administrador



Imatge 5.1.4: Interfícies usuaris

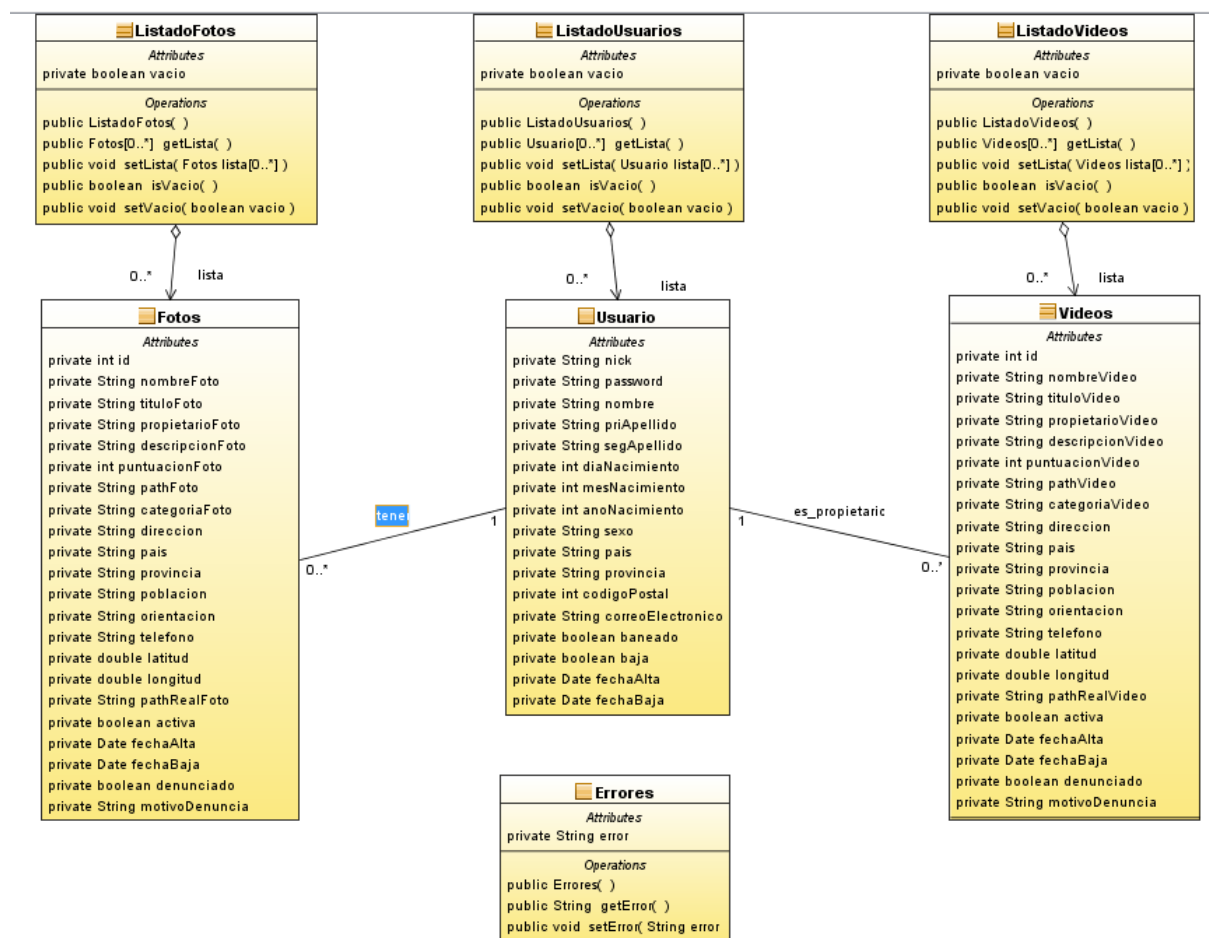
5.2. Domini del projecte

En la *Imatge 5.2.1* podem observar el domini de l'aplicació, són els objectes “Beans” perquè el sistema pugui treballar.

Es pot observar que la classe *Usuario* té una agregació amb *ListadoUsuarios*, aquesta relació serveix perquè l'administrador pugui gestionar tots els usuaris. També observem dues relacions més, una amb *Fotos* i una altra amb *Videos*, aquesta dues relacions serveixen per associar i mostrar qui és el propietari de les fotos i vídeos.

Tant en *Fotos* com en *Videos* podem veure que tenim agregacions a unes llistes, aquestes llistes serviren perquè un usuari qualsevol pugui fer una consulta de categoria, o bé, perquè l'administrador pugui gestionar les fotografies, vídeos dels usuaris.

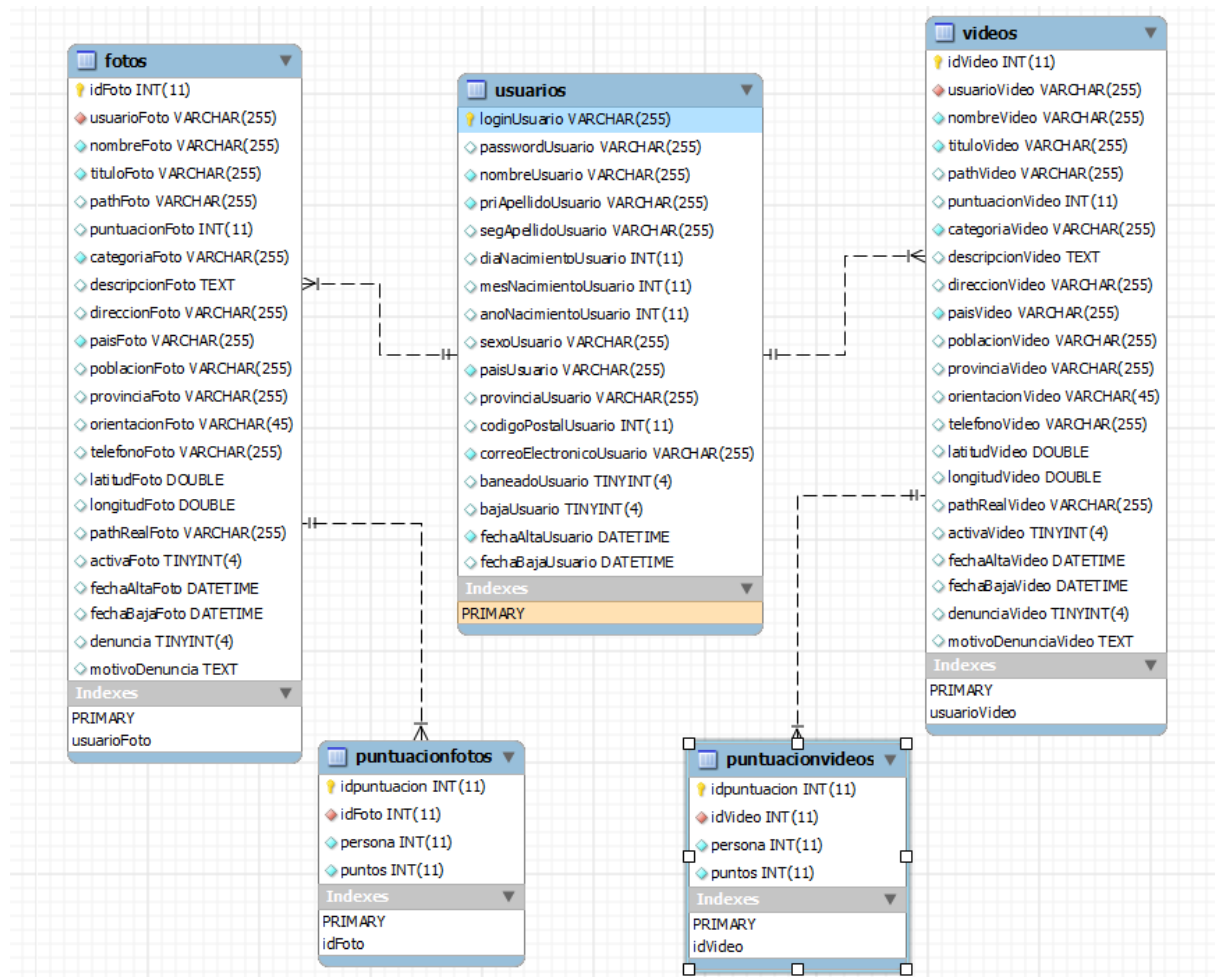
També podem observar una classe que s'anomena *Errores*. Aquesta classe té com a objectiu fer els missatges d'error que es mostraren per pantalla.



Imatge 5.2.1: Model Domini Beans

5.3. Disseny de la base de dades

La *Imatge 5.3.1* és el disseny de la base de dades que guarda la informació del projecte. La taula *Usuario* guarda la informació dels usuaris del sistema i de l'administrador. La taula *Fotos* guarda la informació de la fotografia i l'URL per trobar-la en el FTP, la puntuació actual i les coordenades del Google Maps, però no el número de gent que ha votat la foto d'això s'encarrega la taula *PuntuacionFotos*. El mateix li passa a la taula *Videos* només guarda informació del vídeo, la manera de trobar-ho en el FTP, la puntuació actual i les coordenades en el Google Maps, però el número de gent que ha votat el vídeo està en la taula *PuntuacionVideos*.



Imatge 5.3.1: Diagrama de la base de dades

6. Desenvolupament amb NetBeans i Framework Struts

6.1. Introducció a Struts

Quan es van inventar els Servlets Java, molts programadors es van adonar que eren més ràpids i més potents que el CGI estàndard, portables i extensibles infinitament.

Però escriure infinites sentències `println ()` per enviar HTML al navegador era problemàtic. La resposta van ser les JavaServer Pages, que ens van deixar escriure servlets dins d'elles. Ara els desenvolupadors podien barrejar fàcilment HTML amb codi Java, i tenir tots els avantatges dels servlets.

Les aplicacions web Java es van convertir ràpidament en "centrades-en-JSP". Això, per si sol no era dolent, però no van bé per resoldre problemes de control de flux i altres problemes propis de les aplicacions Web.

Clarament es necessitava un altre model. Alguns desenvolupadors es van adonar que les JavaServer Pages i els servlets es podrien utilitzar junts per desplegar aplicacions web. Els servlets podrien ajudar amb el control de flux, i les JSPs podrien enfocar a escriure HTML. Utilitza JSP i servlets junts s'ha donat a conèixer com el Model 2 (quan usar només JSPs era el Model 1).

El projecte Struts el va llançar el maig de 2000, Craig R. McClanahan per proporcionar un marc de treball MVC estàndard a la comunitat Java.

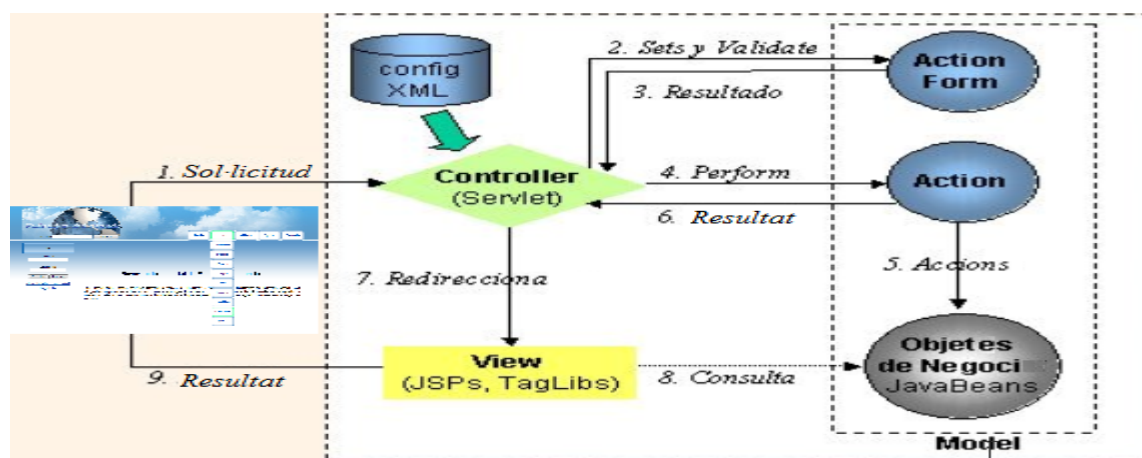
6.2. Què és Struts?

És un framework que implementa el patró d'arquitectura MVC en Java. Un framework és l'extensió d'un llenguatge mitjançant una o més jerarquies de classes que implementen una funcionalitat i que (opcionalment) poden ser esteses. El bastiment pot involucrar TagLibraries.

El patró d'arquitectura MVC (Model-View-Controller) és un patró que defineix l'organització independent del Model (Objectes de Negoci), el View (interfície amb l'usuari o un altre sistema) i el Controller (controlador del workflow de l'aplicació).

6.3. Com funciona els Struts?

El navegador genera una sol·licitud que és atesa pel controlador (un Servlet especialitzat). El mateix s'encarrega d'analitzar la sol·licitud, seguir la configuració que se li ha programat en la seva XML i trucar a l'Action corresponent passant-li els paràmetres enviats. L'Action instància i / o utilitzarà els objectes de negoci per concretar la tasca. Segons el resultat que retorni l'Action, el Controlador derivarà la generació d'interfície a una o més JSPs, les quals podran consultar els objectes del Model per a mostrar informació d'aquests. Representació en la *Imatge 6.3.1*.



Imatge 6.2.1: Funcionament Struts

Vista

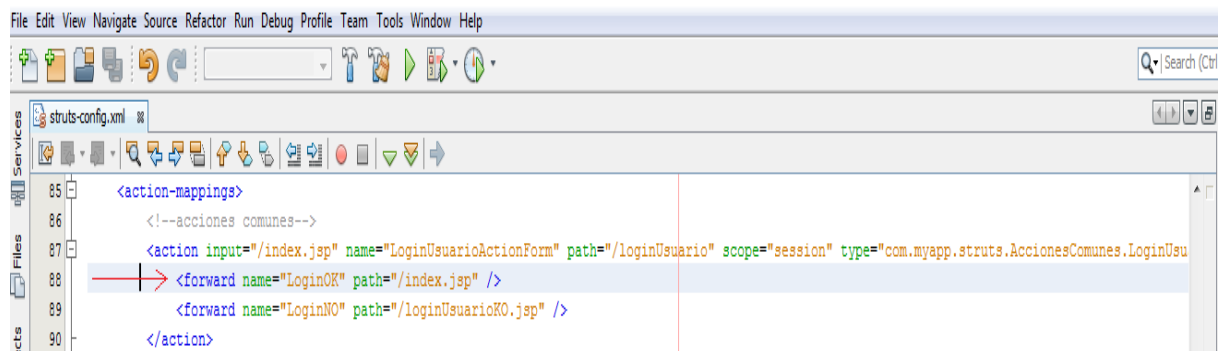
La vista es troba formada per un conjunt de pàgines JSP. Struts proporciona suport per a construir aplicacions multi-idioma, interacció amb formularis i altres utilitats mitjançant la utilització d'etiquetes especials (TagLibraries).

Controlador

El controlador comprèn la funcionalitat involucrada des que un usuari genera una comanda HTTP (clic a un link, enviament d'un formulari, etc.) Fins que es genera la interfície de resposta. Entre mig, cridarà als objectes de negoci del Model perquè resolguin la funcionalitat pròpia de la lògica de negoci i segons el resultat de la mateixa executarà la JSP que hagi de generar la interfície resultant.

Struts inclou un servlet que a partir de la configuració de struts-config.xml rep les sol·licituds de l'usuari, truca al Action Bean que correspongui i, segons el que aquest retorni, executa una JSP.

Classe Action: L'objectiu d'una classe Action és processar una sol·licitud, mitjançant el seu mètode perform (), i retornar un objecte ActionForward que identifica on s'hauria de tornar a enviar el control (per exemple a una JSP) per proporcionar la resposta apropiada. *Imatge 6.2.2.*



```

85 <action-mappings>
86 <!--acciones comunes-->
87 <action input="/index.jsp" name="LoginUsuarioActionForm" path="/loginUsuario" scope="session" type="com.myapp.struts.AccionesComunes.LoginUsu
88 <forward name="LoginOK" path="/index.jsp" />
89 <forward name="LoginNO" path="/loginUsuarioKO.jsp" />
90 </action>

```

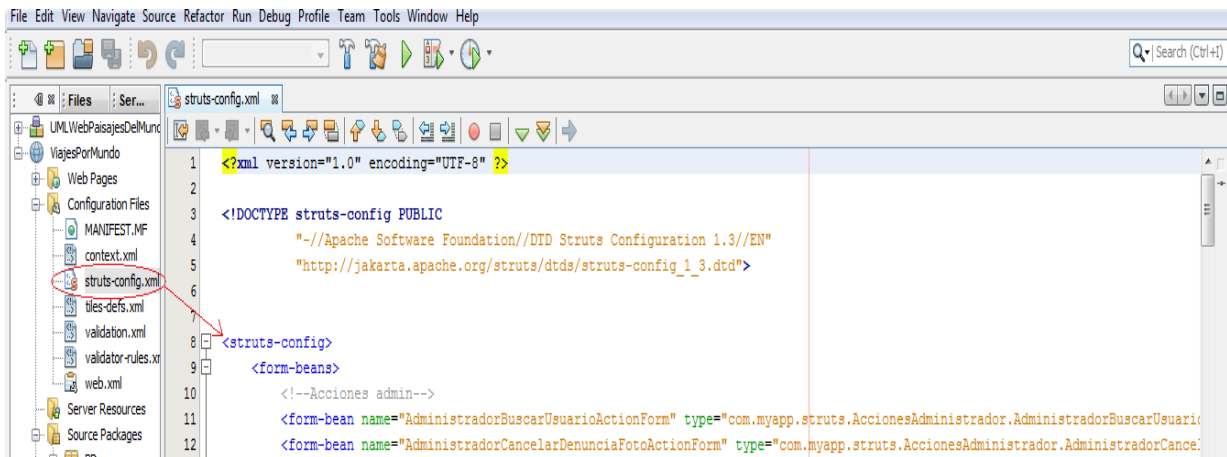
Imatge 6.2.2: Classe ActionFoward, re-direcciona cap a una JSP determinada

➤ **Arxiu Struts-config.xml:** Dins de l'arxiu <struts-config>, hi ha dos elements importants que són usats per descriure les nostres accions:

○ <form-beans>

Aquesta secció conté les nostres definicions de Beans. Fem servir un element <form-bean> per cada bean de formulari, que té els següents atributs importants, *Imatge 6.2.3:*

- name: Un identificador únic per a aquest bean, que serà utilitzat per referenciar en els corresponents mapatges d'accions. Normalment, és també el nom de l'atribut de sol·licitud o sessió sota el qual s'emmagatzema aquest bean de formulari.
- type: El nom totalment qualificat de la classe Java del nostre bean de formulari.

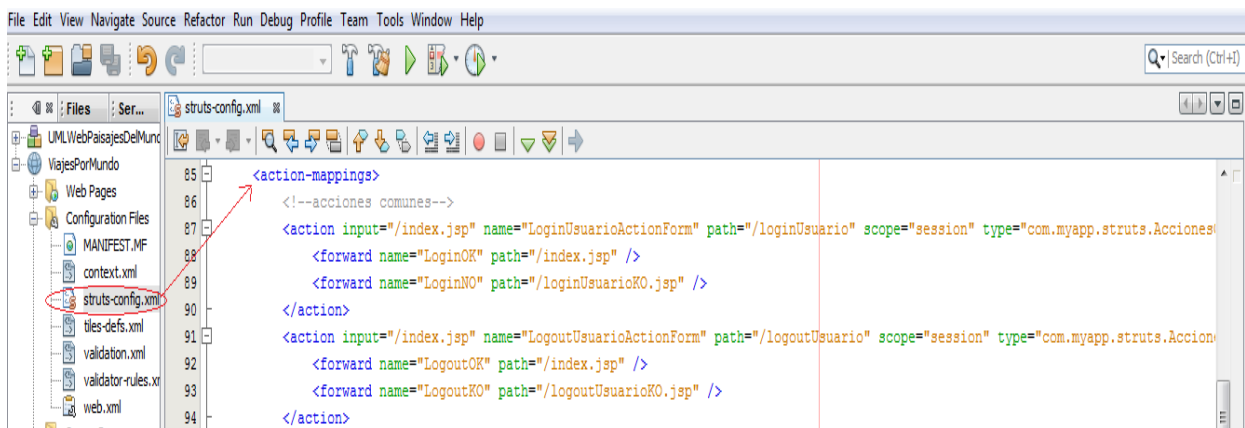


Imatge 6.2.3: Arxiu Struts-config.xml (<form-beans>)

○ <action-mappings>

Aquesta secció conté les nostres definicions d'accions. Fem servir un element <action> per cada una de les nostres accions que vulguem definir. Cada element action requereix que es defineixin els següents atributs, *Imatge 6.2.4*:

- path: El path a la classe action en relació al context de l'aplicació.
- type: El nom totalment qualificat de la classe Java de la nostra classe Action.
- name: El nom del nostre element <form-bean> per utilitzar amb aquesta action.

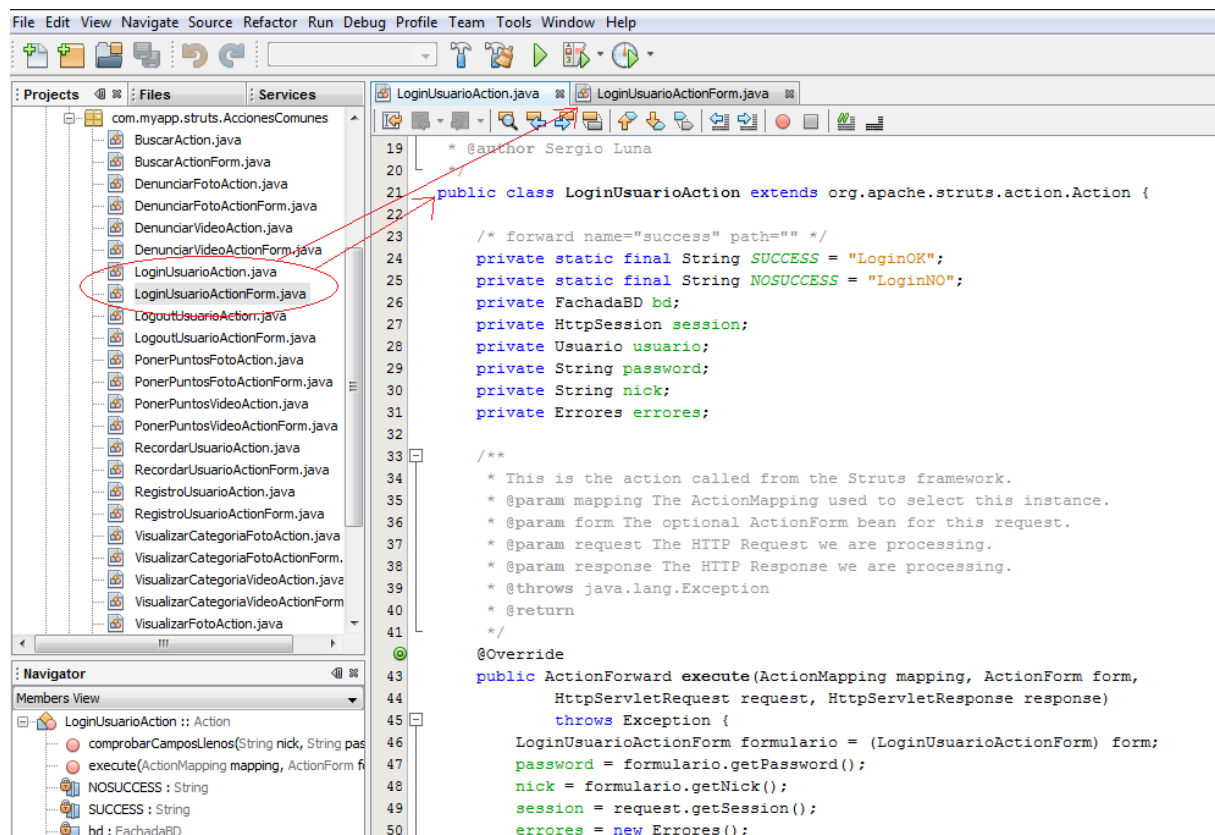


Il·lustració 6.2.4: Arxiu Struts-config.xml (<action-mapping>)

Model

El Model comprèn tots els Objectes de Negoci on s'implementa la lògica de negoci i on ha de suportar tots els requisits funcionals del sistema sense barrejar amb parts corresponents al workflow que corresponen al controlador.

Una de les tasques que durant el desenvolupament d'una aplicació provoca molta feina (encara que en realitat no ho mereixin) és la interacció amb formularis, ja sigui per editar o obtenir nova informació. Les comprovacions, la gestió d'errors, el tornar a presentar-li el mateix formulari a l'usuari amb els valors que va posar i els missatges d'error i un llarg etcètera suporta Struts a finals de fer-nos la vida una mica més fàcil. La idea és la següent: tota la feina de comprovacions i generació de missatges d'error s'implementa en els ActionForm i tot el treball de generació d'interfície en la / les JSP, *Imatge 6.2.5*.



Il·lustració 6.2.5 : Classes Action i ActionForm

6.4. Exemple línia d'execució cas d'ús Login usuari

A continuació s'exposa una línia d'execució d'un cas d'ús, per tal d'il·lustrar tot l'anteriorment explicat.

Introduïm el Nick i Password en el formulari de login *Imatge 6.4.1*.



Imatge 6.4.1: Introducció dades

Després d'acceptar, el sistema anirà cap el strut-config.xml per trobar el "do" corresponent a l'acció de loginUsuario.do, d'aquesta manera el sistema sap a quin Action té que dirigir *Imatge 6.4.2 i Imatge 6.4.3*.

```
<form action="loginUsuario.do">
  <tr>
    <td width="200px" height="50px" background="images/title.gif" align="center">
      <p>Login</p>
    </td>
  </tr>
</form>
```

Imatge 6.4.2: Index.jsp

```

<action-mappings>
  <!--acciones comunes-->
  <action input="/index.jsp" name="LoginUsuarioActionForm" path="/loginUsuario" scope="session" type="com.myapp.struts.AccionesComunes.LoginUsuarioAction" >
    <forward name="LoginOK" path="/index.jsp" />
    <forward name="LoginNO" path="/loginUsuarioKO.jsp" />
  </action>

```

Imatge 6.4.3: Struts-config.xml

Com ja sap el sistema cap a quin Action dirigir-se, comença a fer l'execució per donar un resultat amb l'ActionForward.

```

public ActionForward execute(ActionMapping mapping, ActionForm form,
    HttpServletRequest request, HttpServletResponse response)
    throws Exception {
  LoginUsuarioActionForm formulario = (LoginUsuarioActionForm) form;
  password = formulario.getPassword();
  nick = formulario.getNick();
  session = request.getSession();
  errores = new Errores();

```

Imatge 6.4.5 : Crida al ActionForm

Podem observar que l'Action crida a l'ActionForm *Imatge 6.4.5*, aquesta acció té com a objectiu aconseguir els camps del formulari del JSP gràcies a l'ActionForm (important, per aconseguir els camps del formulari existent en la pàgina JSP, els atributs de l'ActionForm han de coincidir amb els del JSP en el nom) *Imatge 6.4.6* i *Imatge 6.4.7*.

```

public class LoginUsuarioActionForm extends org.apache.struts.action.ActionForm {

  private String nick;
  private String password;

```

Imatge 6.4.6: ActionFrom

```

<tr>
  <td width="200px" height="50px" align="center">
    <p>
      <label>Usuario</label>
    </p>
    <p>
      <input type="text" name="nick"/>
    </p>
  </td>
</tr>
<tr>
  <td width="200px" height="50px" align="center">
    <p><label>Contraseña</label></p>
    <p><input type="password" name="password"/>
    </p>
  </td>
</tr>

```

Imatge 6.4.7: Index.jsp

Després de que l'Action hagi aconseguit les dades de l'ActionForm, comença a cridar els Beans que li faran falta per continuar la seva execució. Com podem observar crida al Bean *Usuario* i crea una instància de la Base de Dades. Després de crear aquesta instància, buscarà en la BD l'usuari (suposarem que si existeix) i el posa en el Bean usuari. Comprova si tot a anat bé i si és així *Imatge 6.4.8*, anirà al strut-config.xml per buscar la situació de "LoginOK" i saber a quin JSP dirigir-se per presentar els resultats *Imatge 6.4.9* i *Imatge 6.4.10*.

```

if (comprobarCamposLlenos(nick, password)) {
  bd = FachadaBD.getInstancia();
  usuario = bd.loggearUsuario(nick, password);
  if (usuario != null) {
    session.setAttribute("usuario", usuario);
    return mapping.findForward(SUCCESS); ← Situació tot OK
  } else {
    errores.setError("Error, el usuario no existe o a sido eliminado por el administrador.");
    session.setAttribute("errores", errores);
    return mapping.findForward(NOSUCCESS);
  }
} else {
  errores.setError("Error, campos vacios para realizar el Login.");
  session.setAttribute("errores", errores);
  return mapping.findForward(NOSUCCESS);
}

```

Imatge 6.4.8: Action

```
<action input="/index.jsp" name="LoginUsuarioActionForm" path="/loginUsuario" scope="session"
  <forward name="LoginOK" path="/index.jsp" />
  <forward name="LoginNO" path="/loginUsuarioKO.jsp" />
</action>
```

Imatge 6.4.9: Struts-config.xml



Imatge 6.4.10: Index.jsp

6.5. Exemple línia d'execució cas d'ús Afegir fotografia

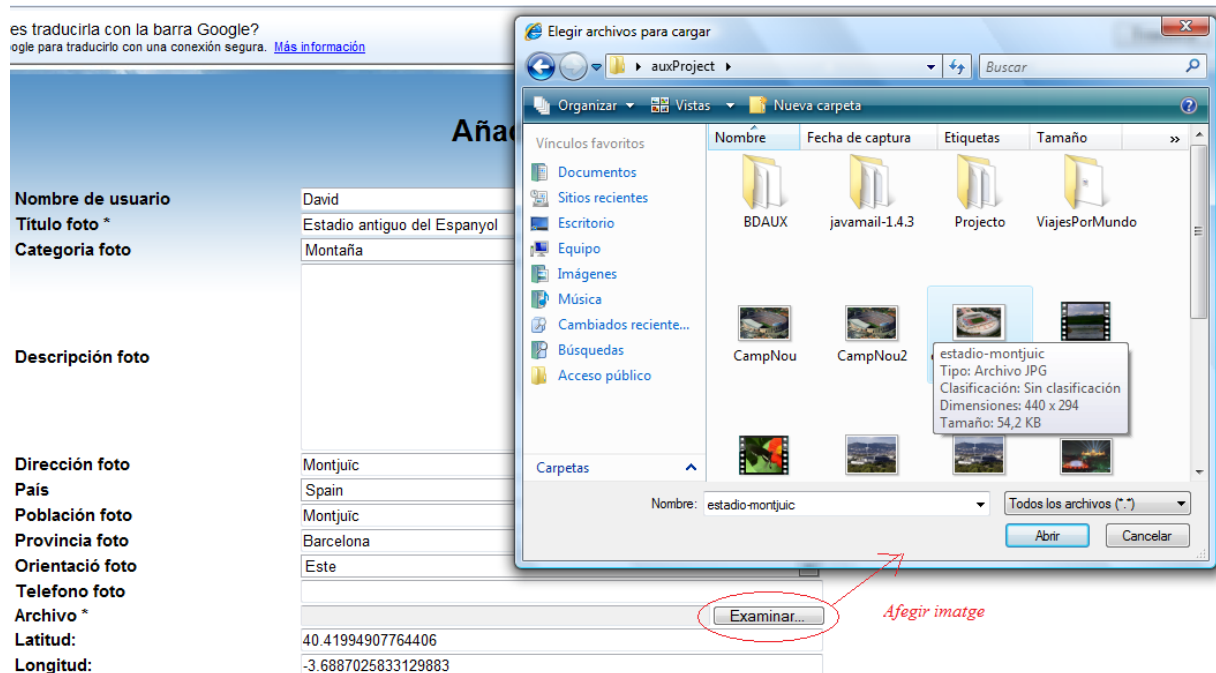
L'usuari ha tingut que fer un login anteriorment per poder fer el cas d'ús que s'explica a continuació.

L'usuari fa clic en el botó del menú privat que posa *Añadir Fotos*. I ens mostrarà un formulari per poder afegir una fotografia en el sistema *Imatge 6.5.1*.

Añadir foto	
Nombre de usuario	David
Título foto *	
Categoría foto	Montaña
Descripción foto	
Dirección foto	
País	Afganistán
Población foto	
Provincia foto	
Orientación foto	Norte
Telefono foto	
Archivo *	<input type="button" value="Examinar..."/>
Latitud:	40.41994907764406
Longitud:	-3.6887025833129883

Imatge 6.5.1 : Formulari per afegir una fotografia

En el formulari de la *Imatge 6.5.1*, s'introdueix tots els detalls de la fotografia (però només són obligatoris els camps marcats amb *). Com podem observar en aquest formulari és obligatori afegir un arxiu que sigui una imatge, l'usuari ha de buscar en el seu sistema la fotografia que es vol afegir a l'aplicació, *Imatge 6.5.2*.

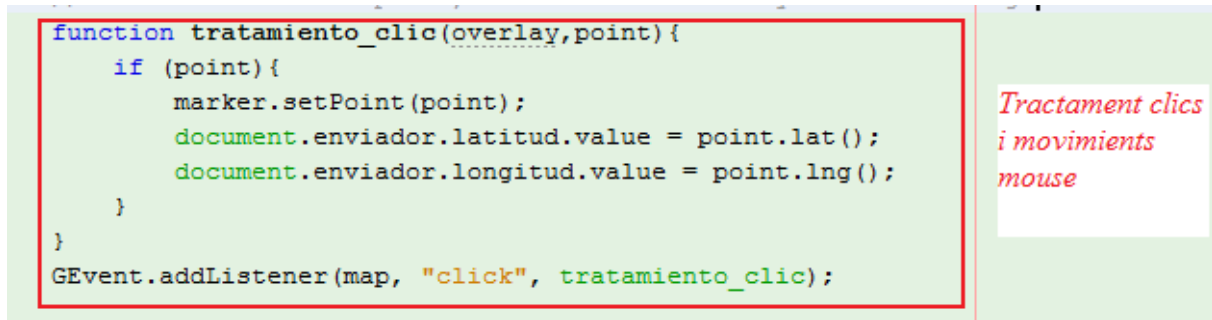


Imatge 6.5.2: Selecció de la imatge que es vol afegir

Després de que s'hagi afegit la imatge en el formulari tindrem que situar aquesta imatge en el mapa del Google Maps, la *Latitud* i la *Longitud* fan referència a la marca situada en el Google Maps. Aquesta marca es té que situar on es va realitzar la fotografia o el més pròxima possible a aquesta posició. En la *Imatge 6.5.3.* podem observar com els camps longitud i latitud canvien a mesura que anem fent clic en el mapa del Google Maps, això és gràcies al script de la *Imatge 6.5.4.* que s'encarrega de recollir les accions del mouse.



Imatge 6.5.3: Posicionament de la Foto en el Google Maps



Imatge 6.5.4: Tractament clics i moviments del mouse

Al fer clic en el “Confirmar” el formulari del JSP *Imatge 6.5.5*, recull les dades i es dirigeix cap el struts-config.xml, per saber a quin Action a d’anar *Imatge 6.5.6*. Quan troba l’Action que realitzar l’acció, comença a fer l’execució per donar un resultat amb l’ActionForward.

```
<form name="enviador" method="post" action="subir.do" enctype="multipart/form-data">
  <table>
    <tr>
      <td width="347" scope="col">
        <div align="left">
          <strong>Nombre de usuario</strong>
        </div>
      </td>
      <td width="452" scope="col">
        <div align="left">
          <input type="text" name="propietario" size="70" value="<c:out value='${usuario.nick}'/>
        </div>
      </td>
    </tr>
  </table>
```

Imatge 6.5.5: usuarioSubirFoto.jsp

```
<action input="/usuarioSubirFoto.jsp" name="UsuarioSubirFotosActionForm" path="/subir" scope="session" type="c">
  <forward name="SubidaFotoOK" path="/usuarioSubirFotoOK.jsp" />
  <forward name="SubidaFotoKO" path="/usuarioSubirFotoKO.jsp" />
</action>
```

Imatge 6.5.6: Struts-config.xml

Podem observar que l’Action crida a l’ActionForm *Imatge 6.5.7*, aquesta acció té com a objectiu aconseguir els camps del formulari del JSP gràcies a l’ActionForm (important, per aconseguir els camps del formulari existent en la pàgina JSP, els atributs de l’ActionForm han de coincidir amb els del JSP en el seu nom) *Imatge 6.5.8 i Imatge 6.5.9*.

```

public ActionForward execute(ActionMapping mapping, ActionForm form,
    HttpServletRequest request, HttpServletResponse response)
    throws Exception {
    UsuarioSubirFotosActionForm formulario = (UsuarioSubirFotosActionForm) form;
    String auxPasarStrings;
    session = request.getSession();
    // Process the FormFile
    FormFile archivo = formulario.getArchivo();
    foto = new Fotos();
    errores = new Errores();
    bd = FachadaBD.getInstancia();
}

```

Crida al ActionForm per aconseguir atributs del JSP

Imatge 6.5.7: Crida a l'ActionForm des de l'Action

```

public class UsuarioSubirFotosActionForm extends org.apache.struts.action.ActionForm {
    private FormFile archivo;
    private String propietario;
    private String tituloFoto;
    private String categoria;
    private String descripcionFoto;
    private String direccionFoto;
    private String paisFoto;
    private String poblacionFoto;
    private String provinciaFoto;
    private String orientacionFoto;
    private String telefonoFoto;
    private double latitud;
    private double longitud;
}

```

Imatge 6.5.8: Atributs ActionForm que haurà de coincidir amb el JSP

```

<form name="enviador" method="post" action="subir.do" enctype="multipart/form-data">
  <table>
    <tr>
      <td width="347" scope="col">
        <div align="left">
          <strong>Nombre de usuario</strong>
        </div>
      </td>
      <td width="452" scope="col">
        <div align="left">
          <input type="text" name="propietario" size="70" value="<@out value='${usuario.nick}'/>" readonly/>
        </div>
      </td>
    </tr>
    <tr>
      <td width="347" scope="col">
        <div align="left">
          <strong>Titulo foto *</strong>
        </div>
      </td>
      <td width="452" scope="col">
        <div align="left">
          <input type="text" name="tituloFoto" size="70"/>
        </div>
      </td>
    </tr>
    <tr>
      <td width="347" scope="col">
        <div align="left">
          <strong>Categoria foto</strong>
        </div>
      </td>
      <td width="452" scope="col">
        <div align="left">
          <select name="categoria" style="width:450px">

```

Imatge 6.5.9: Inputs del usuarioSubirFoto.jsp que han de coincidir amb l'ActionForm

Després de que l'Action hagi aconseguit les dades de l'ActionForm, comença a cridar els Beans que li faran falta per continuar la seva execució. Com podem observar crida al Bean *Foto* i crea una instància de la Base de Dades. Després de crear aquesta instància, buscarà en la BD la foto (suposarem que no existeix), crea l'objecte Bean foto, l'omple amb les dades de l'ActionForm i el guarda en la BD. Comprova si tot a anat bé i si és així *Imatge 6.5.10*, anirà al strut-config.xml per buscar la situació de SubidaFotoOK i saber a quin JSP dirigir-se per presentar els resultats *Imatge 6.5.11 i Imatge 6.5.12*.

```

if (!formulario.getTituloFoto().equalsIgnoreCase("")) {
    if (subirFoto(archivo, foto)) {
        String fileName = archivo.getFileName();
        auxPasarStrings = new String(fileName.getBytes(), "UTF-8");
        fileName = auxPasarStrings;
        foto.setNombreFoto(auxPasarStrings);
        auxPasarStrings = new String(formulario.getTituloFoto().getBytes(), "UTF-8");
        foto.setTituloFoto(auxPasarStrings);
        auxPasarStrings = new String(formulario.getPropietario().getBytes(), "UTF-8");
        foto.setPropietarioFoto(auxPasarStrings);
        auxPasarStrings = new String(formulario.getDescripcionFoto().getBytes(), "UTF-8");
        foto.setDescripcionFoto(auxPasarStrings);
        auxPasarStrings = new String(formulario.getCategoria().getBytes(), "UTF-8");
        foto.setCategoriaFoto(auxPasarStrings);
        foto.setPuntuacionFoto(0);
        foto.setPathFoto("./paisajes/Fotos/" + fileName);
        auxPasarStrings = new String(formulario.getPoblacionFoto().getBytes(), "UTF-8");
        foto.setPoblacion(auxPasarStrings);
        auxPasarStrings = new String(formulario.getProvinciaFoto().getBytes(), "UTF-8");
        foto.setProvincia(auxPasarStrings);
        auxPasarStrings = new String(formulario.getDireccionFoto().getBytes(), "UTF-8");
        foto.setDireccion(auxPasarStrings);
        auxPasarStrings = new String(formulario.getPaisFoto().getBytes(), "UTF-8");
        foto.setPais(auxPasarStrings);
        auxPasarStrings = new String(formulario.getOrientacionFoto().getBytes(), "UTF-8");
        foto.setOrientacion(auxPasarStrings);
        auxPasarStrings = new String(formulario.getTelefonoFoto().getBytes(), "UTF-8");
        foto.setTelefono(auxPasarStrings);
        foto.setLatitud(formulario.getLatitud());
        foto.setLongitud(formulario.getLongitud());
        foto.setFechaAlta(new java.sql.Date(System.currentTimeMillis()));
        foto.setActiva(true);
        if (bd.insertarFoto(foto)) {
            session.setAttribute("foto", foto);
            return mapping.findForward(SUCCESS);
        } else {
            errores.setError("Error al introducir en la BD.");
        }
    }
}

```

→ Situació tot a anat bé

Imatge 6.5.10: Action

```

<action input="/usuarioSubirFoto.jsp" name="UsuarioSubirFotosActionForm" path="/subir" scope="session"
    <forward name="SubidaFotoOK" path="/usuarioSubirFotoOK.jsp" />
    <forward name="SubidaFotoKO" path="/usuarioSubirFotoKO.jsp" />
</action>

```

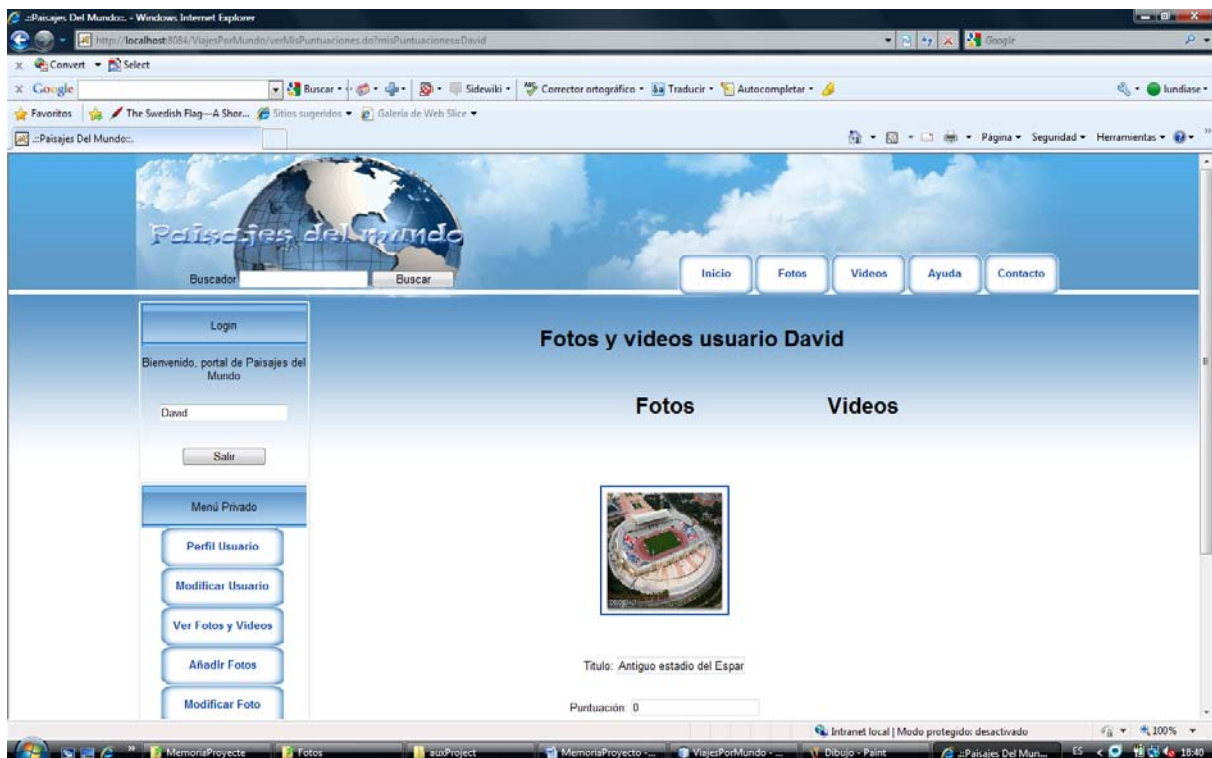
→ Situació Ok

Imatge 6.5.11: Struts-config.xml

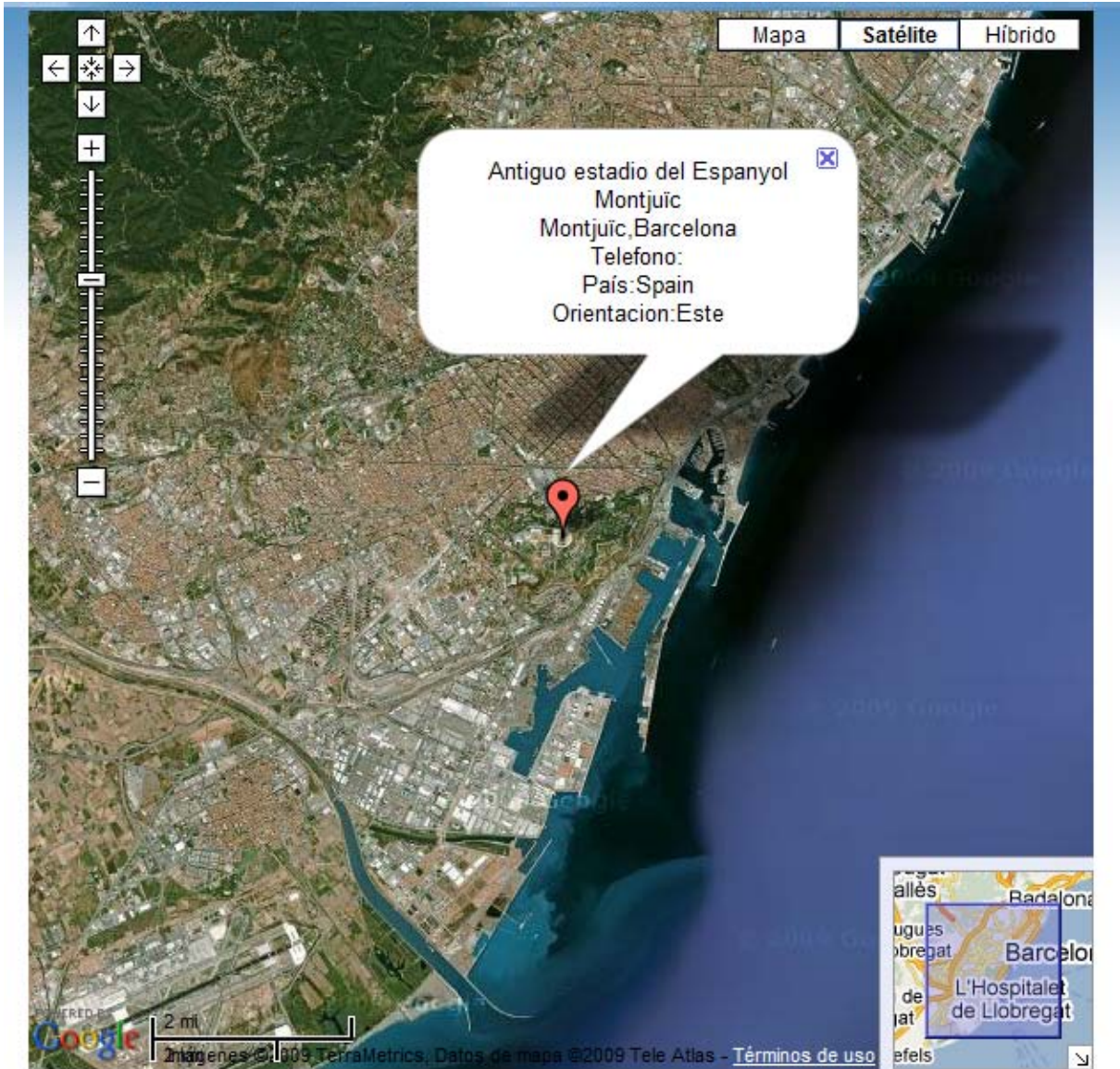


Imatge 6.5.12: JSP Confirmació foto allotjada correctament

Per comprovar si l'usuari realment a allotjat correctament la fotografia es pot fer de dues maneres. Una fer clic en fotos i la categoria amb la que s' ha allotjat, o bé fer clic en "Ver Fotos y Videos" del menú privat *Imatge 6.5.13*. I comprovar la situació de la foto en el Google Maps *Imatge 6.5.14*.



Imatge 6.5.13: Fotos i vídeos d'un usuari



Imatge 6.5.14: Situació de la nova foto en el Google Maps

7. API Google Maps



En aquest apartat es descriuen les funcions de l'API de Google Maps que s'utilitza en el desenvolupament del projecte. S'explicaran les funcions més utilitzades per programar la part del mapa. No és un manual de referència sobre la utilització de l'API de Google Maps, donat que pot haver-hi centenars de maneres de treballar amb ell.

Primer, abans de començar a treballar hem de saber uns conceptes sobre l'API del Google Maps:

- Per poder manipular les classes de l'API s'ha de fer a través del llenguatge Javascript. Per tant, ja es pot saber que com a mínim dintre de l'etiqueta `<head>` `</head>` sortirà una etiqueta `<script>` `</script>`.
- El servei de mapes de Google Maps és gratuït, però necessita una clau d'accés per poder utilitzar els seus mapes en una aplicació web. Aquesta clau la proporciona Google Code⁶ mitjançant una inscripció, en aquesta inscripció només s'ha d'introduir un correu i la direcció web en la que s'utilitzarà els mapes. En el nostre projecte serà `http://localhost/`.

Automàticament Google et facilita un codi d'accés per poder començar a utilitzar l'API.

- Aquest codi s'ha d'introduir en un script en el JSP que visualitzarà els mapes. El script és el següent:

```
<script src="http://maps.google.com/maps?file=api&v=2&sensor=true_or_false&key= AQUI POSAR EL CODI FACILITAT PER GOOGLE" type="text/javascript"></script>
```

- El següent pas, és mostrar el mapa en la web. Per aconseguir això, s'ha d'utilitzar l'etiqueta <div id="mapa"> </div> que la situarem dintre del codi html de l'aplicació web, just on volem que es visualitzi el mapa. Com tota etiqueta "tag" té atributs de dimensions, alineació, etc... que es poden modificar per ajustar el disseny del mapa al disseny de l'aplicació web. L'etiqueta tindria el format següent o semblant:

```
<div id="map" align="center" style="width: 600px; height: 600px"></div>
```

- Un cop situat l'etiqueta dins de la pàgina desitjada, s'ha de crear un script que provoqui cada vegada que es carregui aquesta pàgina en concret, carregui el mapa automàticament.

Per poder aconseguir aquest objectiu primer s'ha de modificar el codi html de la pàgina que carrega el mapa, això es pot aconseguir posant dins de l'etiqueta <body> </body> una crida a un script "load()" de carrega del mapa, de la següent manera:

```
<body id="body" onload="load()" onunload="GUnload()" text="UTF-8">
```

A continuació, s'ha de crear el script "load()" per tal de que carregui el mapa, amb els valors que volem per defecte sobre el mapa a visualitzar.

```
<script>
  //<![CDATA[
  function load() {
```

Es comprova si el navegador que s'utilitza és compatible amb Google Maps, amb el mètode de l'API de Google Maps, `GBrowserIsCompatible()`.

```
if (GBrowserIsCompatible()) {
```

S'ha de crear una instància de la classe de `GMap2` de Google Maps. I es guarda a una variable “*map*” per treballar amb ella.

```
var map = new GMap2(document.getElementById("map"));
```

Es situa el mapa en una posició inicial i un zoom inicial. La funció `GLatLng()`, és el constructor de coordenades del Google Maps i el 40.4199 i -3.6887 correspon a la latitud i longitud de la situació de Madrid, que és el punt inicial que s'ha decidit partir la cerca. Es centra el mapa en “*Madrid*” i amb un zoom de 7.

```
var madrid = new GLatLng(40.41994907764406, -3.6887025833129883);  
map.setCenter(madrid, 7);
```

S'ha de carregar el tipus de mapa inicial. El mapa s'ha posat en tipus de visualització en satèl·lit, però també es pot posar en `G_NORMAL_MAP` que posa la visualització com un mapa de carrers o bé en `G_HIBRYD_MAP` que és una barreja dels dos tipus anteriors.

```
map.setMapType(G_SATELLITE_MAP);
```

Es carrega els controls del mapa, l'escala que volem i una miniatura en la part dreta del mapa. Amb aquest controls l'usuari pot navegar i moure's pel mapa del Google Maps.

```
map.addControl(new GMapTypeControl());  
map.addControl(new GLargeMapControl());  
map.addControl(new GScaleControl());  
map.addControl(new GOverviewMapControl());
```

Si volem que l'usuari pugui interactuar amb el ratolí per situar una marca en el mapa. Primer s'ha de capturar la posició i el zoom inicial del mapa, i amb l'ajuda d'una funció createMarker() es crea una primera marca en el centre, perquè aquesta sigui manipulada més endavant.

```
var center = map.getCenter();  
document.enviador.latitud.value = center.lat();  
document.enviador.longitud.value = center.lng();  
  
function createMarker(point, icon, tag) {  
    var marker = new GMarker(point, icon);  
    GEvent.addListener(marker, "click", function() {  
        marker.openInfoWindowHtml(tag);  
    });  
    return marker;  
}  
  
var tag = "";  
var marker = createMarker(madrid, "", tag);  
map.addOverlay(marker);
```

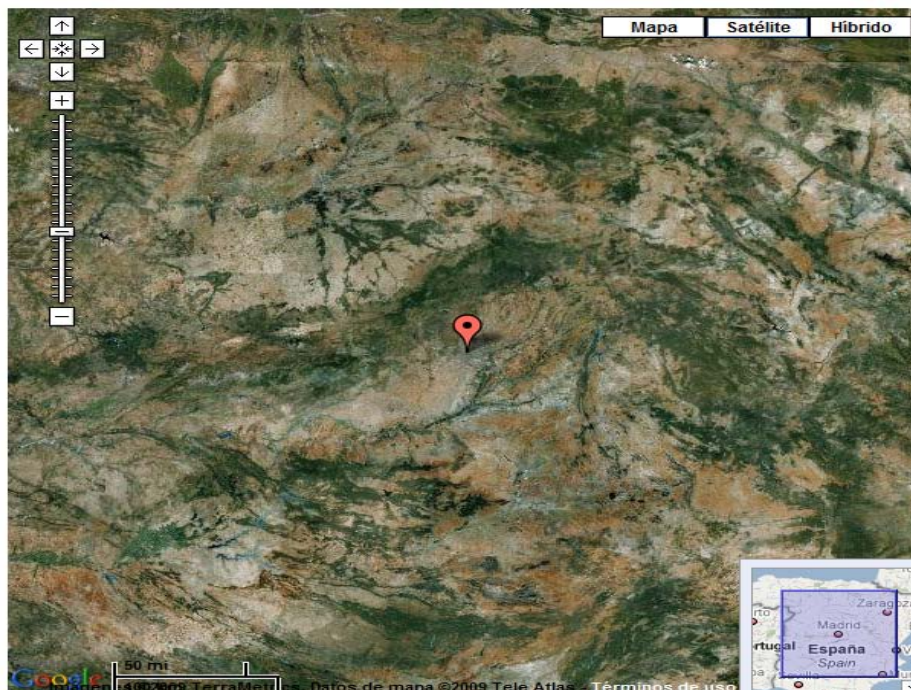
Si el mapa ha d'interactuar amb l'usuari, s'ha d'incloure una nova funció dins del mateix script (prèviament s'ha executat la funció `createMarker()`) amb el nom de `tratamiento_clic()`. Aquesta funció escolta les possibles accions que realitza l'usuari amb el ratolí en el mapa i realitza un tractament determinat.

```
function tratamiento_clic(overlay,point){
  if (point){
    marker.setPoint(point);
    document.enviador.latitud.value = point.lat();
    document.enviador.longitud.value = point.lng();
  }
}
GEvent.addListener(map, "click", tratamiento_clic);
```

En cas contrari, només s'ha de finalitzar el script.

```
    }
  }
//]]>
</script>
```

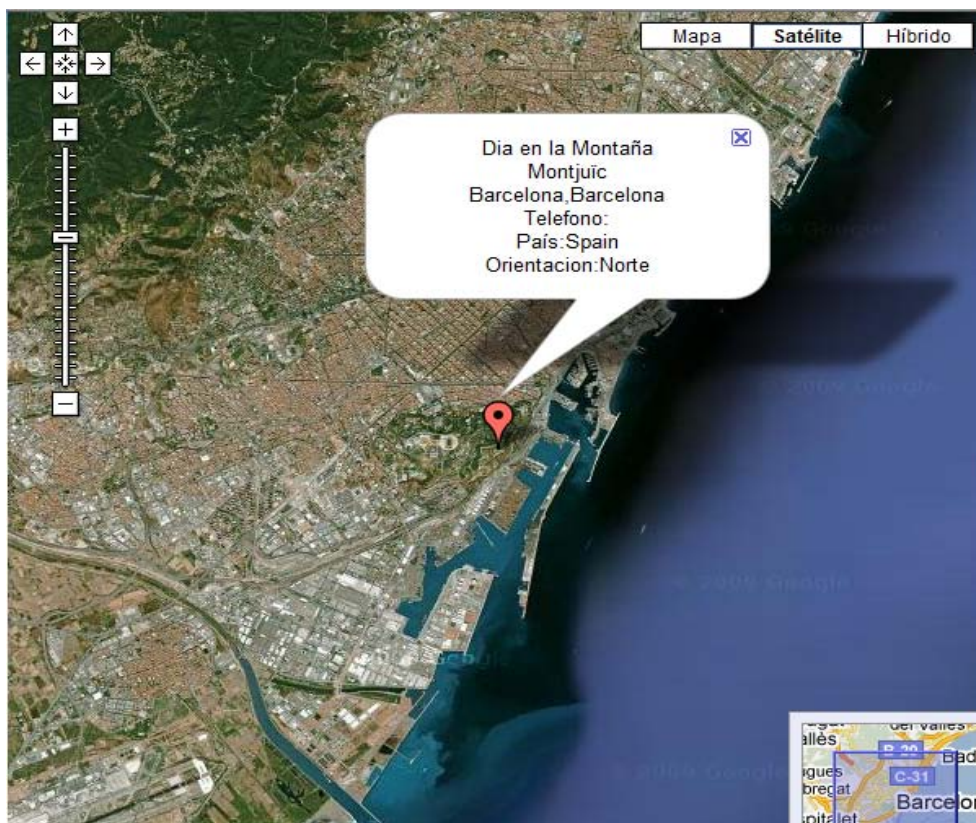
Aquest és el resultat que obtenim del script creat anteriorment.



Si també volem afegir informació a les marques que posicionen a les fotografies o vídeos s'ha d'afegir el següent codi dintre del script:

```
var tag = '<spam style="font-size: 10pt; font-family: verdana; text-decoration:underline; font-style:italic">${foto.tituloFoto}</spam>\n\<br />\n\<spam style="font-size: 8pt; font-family: verdana">${foto.direccion}<br />\n\<n\> ${foto.poblacion}, ${foto.provincia}<br />Telefono: ${foto.telefono}<br />\n\<n\> Pais: ${foto.pais}<br />Orientacion: ${foto.orientacion}</spam>';  
var marker = createMarker(posi, "", tag);  
map.addOverlay(marker);
```

L'efecte obtingut sobre la marca en el mapa és el següent:



8. FormFile i File

En aquest apartant, té com a objectiu explicar punts del codi font de l'aplicació web que han sigut especialment difícils de desenvolupar i que poden ser útils per a pròxims projectes.

8.1. Afegir un arxiu a un FTP

Aquest problema està pensat sobre un FTP simulat en l'ordinador que fa de servidor, tot i això també es pot aplicar a un FTP d'un Host en Internet.

El primer pas, condicionar el formulari del JSP perquè pugui recollir arxius i l'aplicació web els reconegui com a tal. Aquest condicionament s'ha de realitzar com s'observa en la *Imatge 8.1.1*.

```
<form name="enviador" method="post" action="subir.do" enctype="multipart/form-data">

<tr>
  <td width="347" scope="col">
    <div align="left">
      <strong>Archivo *</strong>
    </div>
  </td>
  <td width="452" scope="col">
    <div align="left">
      <input type="file" name="archivo" size="54"/>
    </div>
  </td>
</tr>

</form>
```

Imatge 8.1.1: Formulari per afegir un fitxer

Amb el “enctype=multipart/form-data” s’ha aconseguit que el formulari interpreti el “input type=file” com a un fitxer, però encara no es pot saber quin tipus de fitxer s’està intentant afegir al FTP.

El següent pas és preparar l’ActionForm per rebre un fitxer. L’atribut que fa referència al fitxer que s’ha d’afegir al FTP, no s’ha d’interpretar com un String (tot i que surt un URL absolut de com arribar a trobar el fitxer), s’ha de declarar com un FormFile⁷, *Imatge 8.1.3*. Per poder obtenir aquest tipus de fitxer s’ha de realitzar un import *Imatge 8.1.2*.

```
import org.apache.struts.upload.FormFile;
```

Imatge 8.1.2: Import llibreria Struts

```
public class UsuarioSubirFotosActionForm extends org.apache.struts.action.ActionForm {  
  
    private FormFile archivo;  
    private String propietario;  
    private String tituloFoto;  
    private String categoria;  
    private String descripcionFoto;  
    private String direccionFoto;  
    private String paisFoto;  
    private String poblacionFoto;  
    private String provinciaFoto;  
    private String orientacionFoto;  
    private String telefonoFoto;  
    private double latitud;  
    private double longitud;  
  
    /**  
     * @return Returns the theFile.  
     */  
    public FormFile getArchivo() {  
        return archivo;  
    }  
}
```

Imatge 8.1.3: ActionForm per afegir un fitxer

I per últim, s'ha de fer en l'Action un procés per transformar el FormFile en un File⁸. Com en el cas anterior s'ha de fer imports de la llibreria de Struts i també de la llibreria de java.io. S'implementa una funció dintre de l'Action que s'encarrega de realitzar a l'operació de passar el FormFile a File, *Imatge 8.1.3*.

```
import org.apache.struts.upload.FormFile;
import java.io.*;

private boolean subirFoto(FormFile myFile, Fotos foto) throws FileNotFoundException, IOException {
    boolean subido = false;
    String contentType = myFile.getContentType();
    String fileName = myFile.getFileName();
    int fileSize = myFile.getFileSize();
    byte[] fileData = myFile.getFileData();

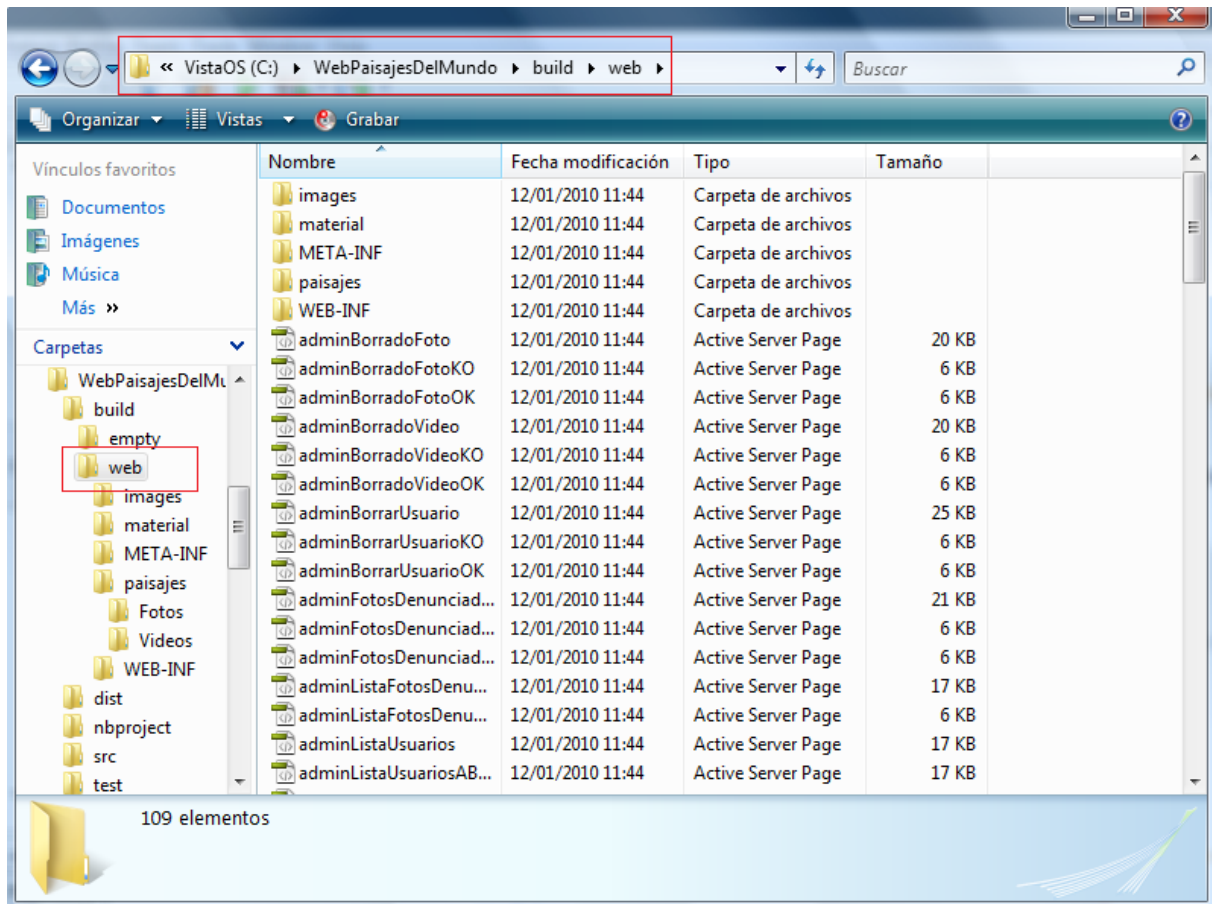
    String filePath = getServlet().getServletContext().getRealPath("/") + "paisajes/Fotos";
    /* Save file on the server */
    if (!fileName.equals("")) {
        File finalFile = new File("C:/WebPaisajesDelMundo/web/paisajes/Fotos/" + fileName);
        File theFile = new File(filePath, myFile.getFileName());
        if (!finalFile.exists()) {
            if (acceptFormat(finalFile, foto)) {
                if (acceptTamano(finalFile)) {
                    finalFile.createNewFile();
                    FileOutputStream fileOutputStream = new FileOutputStream(finalFile);
                    FileOutputStream fileOutputStream2 = new FileOutputStream(theFile);
                    fileOutputStream.write(myFile.getFileData());
                    fileOutputStream2.write(myFile.getFileData());

                    fileOutputStream.close();
                    fileOutputStream2.close();
                    subido = true;
                    foto.setPathRealFoto("C:/WebPaisajesDelMundo/web/paisajes/Fotos/" + fileName);
                } else {
                    return false;
                }
            } else {
                subido = false;
            }
        } else {
            subido = false;
        }
    }
    return subido;
}
```

Imatge 8.1.3: Funció que afegeix un fitxer al FTP

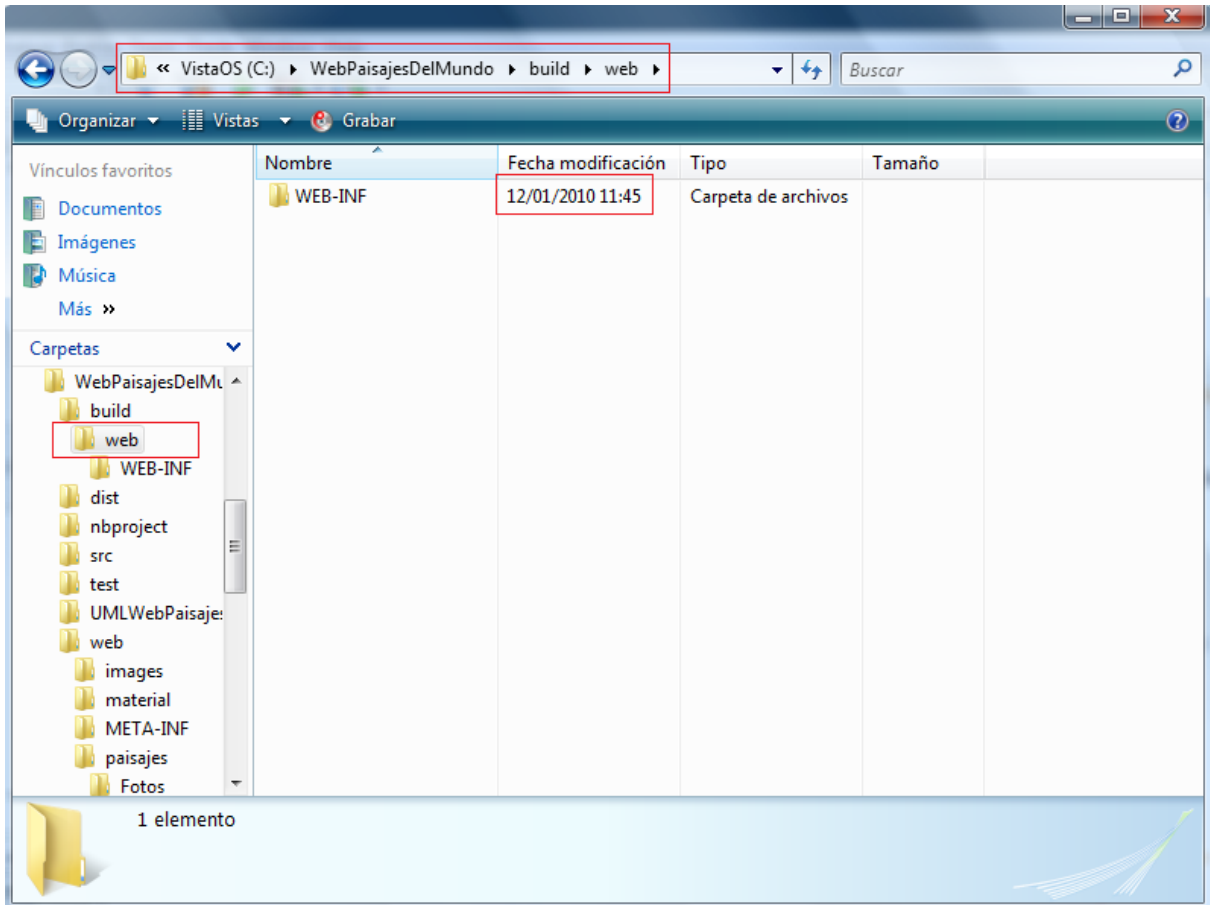
8 <http://java.sun.com/j2se/1.4.2/docs/api/java/io/File.html>

Com s'observa en l' *Imatge 8.1.3* hi ha dos Files. Un s'ha creat a partir de la sentència `new File(getServlet().getServletContext().getRealPath("/")+"paisajes/Fotos",myFile.getFileNme())`. Aquesta sentència afegeix el fitxer en el FTP del Host correctament (si prèviament s'ha creat la carpeta `paisajes/Fotos` en el FTP) *Imatge 8.1.4*.



Imatge 8.1.4. Carpeta Build (execució temporal) del Projecte

El problema apareix quan es treballa de manera local, la sentència funciona correctament i tota l'aplicació web es comporta amb normalitat, si un usuari afegeix una foto nova o un nou vídeo el sistema ho reconeix i el visualitza amb normalitat. Però si es fa clic en “Clean and Rebuild the Project” botó del NetBeans, la foto o vídeo que s'acaba d'afegir desapareix, la informació i la situació en el Google Maps és correcte però no es visualitza la imatge. Aquest problema és perquè el NetBeans elimina la carpeta *build* del projecte i el fitxer s'havia afegit en aquesta carpeta, *Imatge 8.1.5*.



Imatge 8.1.5. Carpeta Build (execució temporal) del Projecte després de fer el Clean

Solució, crear un segon fitxer com s'observa en la *Imatge 8.1.3* que afegeix el fitxer en un URL absolut on està situat el projecte físicament. D'aquesta manera sempre que es fa clic en "Clean and Rebuild the Project" no desapareix el fitxer, perquè realitza una copia en el moment del Rebuild de totes les fotografies i vídeos en la carpeta d'execució.

9. Proves

Per fer les proves, s'han realitzat diferents execucions en els dos navegadors més utilitzats en Internet, Internet Explorer i Firefox Mozilla. Aquestos dos navegadors es queden més del 90% de la quota de mercat. Segons l'estadística entre el 8 de desembre i el 10 de gener realitzada per StatsCounter⁹.

L'entorn de proves consta d'un PC amb les següents característiques:

Sistema Operatiu Windows 7.

Navegadors Internet Explorer 8 i FireFox 3.X.

Realitzarem tres paquets de proves. Una per cada rol d'usuari que existeix en el sistema, per saber quina reacció fa l'aplicació web. I la seva visualització correcte en els diferents navegadors, per tal de trobar falles.

9 <http://gs.statcounter.com/>

En la següent taula *Imatge 9.1*, es poden veure amb detall les proves realitzades a un usuari públic.

Proves realitzades	Finalització Esperada	Navegadors	Resultat Esperat
Visualitzar foto	Mostrar Fotografia i la seva informació	Internet Explorer	Si
		Firefox Mozilla	Si
Visualitzar vídeo	Mostrar Vídeo i la seva informació	Internet Explorer	Si
		Firefox Mozilla	Si
Utilitzar el buscador	Trobar la fotografia o vídeo que es buscava	Internet Explorer	Si
		Firefox Mozilla	Si
Utilitzar l'ajuda de l'aplicació web	Trobar informació necessària	Internet Explorer	Si
		Firefox Mozilla	Si
Contactar amb l'administrador	Enviar un e-mail a l'administrador	Internet Explorer	Si
		Firefox Mozilla	Si
Puntuar foto o vídeo	Posar la puntuació desitjada a una imatge	Internet Explorer	Si
		Firefox Mozilla	Si
Denunciar foto o vídeo	Posar una denúncia i un motiu a una imatge	Internet Explorer	Si
		Firefox Mozilla	Si

Imatge 9.1: Taula de proves per part de l'usuari d'accés públic

En la següent taula *Imatge 9.2*, es poden veure amb detall les proves realitzades a un usuari registrat.

Proves realitzades	Finalització Esperada	Navegadors	Resultat Esperat
Login	Iniciar Sessió	Internet Explorer	Si
		FireFox Mozilla	Si
Visualitzar foto	Mostrar Fotografia i la seva informació	Internet Explorer	Si
		FireFox Mozilla	Si
Visualitzar vídeo	Mostrar Vídeo i la seva informació	Internet Explorer	Si
		FireFox Mozilla	Si
Utilitzar el buscador	Trobar la fotografia o vídeo que es buscava	Internet Explorer	Si
		FireFox Mozilla	Si
Utilitzar l'ajuda de l'aplicació web	Trobar informació necessària	Internet Explorer	Si
		FireFox Mozilla	Si
Contactar amb l'administrador	Enviar un e-mail a l'administrador	Internet Explorer	Si
		FireFox Mozilla	Si
Puntuar foto o vídeo	Posar la puntuació desitjada a una imatge	Internet Explorer	Si
		FireFox Mozilla	Si
Denunciar foto o vídeo	Posar una denuncia i un motiu a una imatge	Internet Explorer	Si
		FireFox Mozilla	Si
Afegir una foto	Fotografia carregada en el sistema	Internet Explorer	Si
		FireFox Mozilla	Si
Afegir un vídeo	Vídeo carregat en el sistema	Internet Explorer	Si
		FireFox Mozilla	Si
Modificar foto	La foto s'ha modificat i s'ha carregat en el sistema	Internet Explorer	Si
		FireFox Mozilla	Si

Modificar vídeo	El vídeo s'ha modificat i s'ha carregat en el sistema	Internet Explorer	Si
		Firefox Mozilla	Si
Eliminar foto	L'usuari ha eliminat del sistema la foto desitjada	Internet Explorer	Si
		Firefox Mozilla	Si
Eliminar vídeo	L'usuari ha eliminat del sistema el vídeo desitjat	Internet Explorer	Si
		Firefox Mozilla	Si
Modificar usuari	L'usuari ha pogut modificar el seu perfil i guardar-ho	Internet Explorer	Si
		Firefox Mozilla	Si
Consultar fotos i vídeos	L'usuari ha pogut visualitzar les seves fotos i vídeos	Internet Explorer	Si
		Firefox Mozilla	Si
Consultar perfil d'usuari	L'usuari ha pogut visualitzar tota la informació del seu perfil	Internet Explorer	Si
		Firefox Mozilla	Si

Imatge 9.2: Taula de proves per part de l'usuari registrat

En la següent taula *Imatge 9.3*, es poden veure amb detall les proves realitzades a l'administrador.

Proves realitzades	Finalització Esperada	Navegadors	Resultat Esperat
Login	Iniciar Sessió com administrador	Internet Explorer	Si
		Firefox Mozilla	Si
Visualitzar foto	Mostrar Fotografia i la seva informació	Internet Explorer	Si
		Firefox Mozilla	Si
Visualitzar vídeo	Mostrar Vídeo i la seva informació	Internet Explorer	Si
		Firefox Mozilla	Si

Utilitzar el buscador	Trobar la fotografia o vídeo que és buscava	Internet Explorer	Si
		Firefox Mozilla	Si
Utilitzar l'ajuda de l'aplicació web	Trobar informació necessària	Internet Explorer	Si
		Firefox Mozilla	Si
Modificar foto Administrador	La foto s'ha modificat i s'ha carregat en el sistema	Internet Explorer	Si
		Firefox Mozilla	Si
Modificar vídeo Administrador	El vídeo s'ha modificat i s'ha carregat en el sistema	Internet Explorer	Si
		Firefox Mozilla	Si
Eliminar foto Administrador	L'administrador ha eliminat del sistema la foto desitjada	Internet Explorer	Si
		Firefox Mozilla	Si
Eliminar vídeo Administrador	L'administrador ha eliminat del sistema el vídeo desitjat	Internet Explorer	Si
		Firefox Mozilla	Si
Eliminar usuari	L'administrador ha pogut eliminar del sistema i les seves imatges associades	Internet Explorer	Si
		Firefox Mozilla	Si
Llistar usuaris	L'administrador ha pogut visualitzar tots els usuaris del sistema i les seves imatges	Internet Explorer	Si
		Firefox Mozilla	Si
Consultar fotos denunciades i cancel·lar denuncia	L'administrador ha pogut consultar les fotos denunciades i cancel·lar la denuncia d'una foto desitjada	Internet Explorer	Si
		Firefox Mozilla	Si
Consultar fotos denunciades i eliminar foto	L'administrador ha pogut consultar les fotos denunciades i eliminar una foto desitjada	Internet Explorer	Si
		Firefox Mozilla	Si

Consultar fotos denunciades i cancel·lar denuncia	L'administrador ha pogut consultar els vídeos denunciats i cancel·lar la denuncia d'un.	Internet Explorer	Si
		Firefox Mozilla	Si
Consultar fotos denunciades i eliminar foto	L'administrador ha pogut consultar els vídeos denunciats i eliminar un desitjat	Internet Explorer	Si
		Firefox Mozilla	Si

Imatge 9.2: Taula de proves per part de l'administrador

10. Ampliacions i aplicacions del projecte

Aquest projecte té infinites ampliacions i aplicacions. A continuació es farà primer un anàlisi de les possibles ampliacions del projecte i integracions en altres projectes, i en segon lloc, un anàlisi de les possibles aplicacions de cara al mercat.

10.1. Ampliacions del projecte

Com a possible ampliació podria ser la introducció d'un sistema d'excursions dels usuaris registrats.

A les excursions se li podran associar fotos o vídeos del trajecte i agafa el recolzament en el Google Maps per generar-la i situar-les en una posició exacta, perquè els usuaris (registrats o no) que visualitzin aquestes excursions, es puguin fer una idea que és trobaren si realitzen aquesta excursió.

Aquesta proposta és realment interessant de desenvolupar, perquè exprimiria al 100% l'API de Google Maps i el projecte es transformaria en un punt de referència pels usuaris que li agraden les excursions, la naturalesa i descobrir nous paratges.

10.2. Aplicacions del projecte

Aquest projecte presenta múltiples aplicacions de cara a ser comercialitzat. Es farà un anàlisi de les dos més importants i més rellevants.

La primera podria ser ampliar la web d'una agència de viatges amb la incorporació del projecte. El projecte seria molt interessant per incorporar-ho a una pàgina d'aquest tipus, perquè seria una manera d'augmentar els beneficis (vendes de viatges) de l'agència. Però hauria de tenir en compte els següents aspectes:

- Primer es tindrà que modificar el projecte per adaptar-la a la presentació, visualització i opcions en les que treballa la web de l'agència.
- Segon, seria molt valuós que es realitzes un estudi previ de les accions que més sovint realitzen els usuaris. Per tal, de modificar el projecte per poder facilitar aquestes accions habituals els usuaris (creant accessos directes) en la unió de les dues webs.

Amb la incorporació del projecte a una pàgina web d'una agència de viatges, l'usuari que volgués contractar un viatge a un país, podrà consultar les experiències d'altres usuaris i els paisatges que podrà trobar en aquell lloc. Això podria ser un "handicap" per decidir-se per aquest viatge.

Una altra podria ser (i més atractiva donat l'explosió de la telefonia mòbil), adaptar la web per un dispositiu mòbil. Els usuaris estarien de viatge o d'excursió amb el seu mòbil amb càmera, i en un moment determinat l'usuari realitza una fotografia o un vídeo d'un paratge molt interessant, i vol compartir aquesta imatge amb altres usuaris. Llavors es connecta a l'aplicació web amb el seu mòbil i afegeix la foto o vídeo en la galeria adient.

Però per poder realitzar aquesta aplicació es tindria en compte els següents aspectes:

- Realitzar un estudi per modificar l'aplicació web i adaptar-la a dispositius mòbils, per exemple, canviar el Google Maps per el Google Maps Mobile¹⁰.
- Pensar quins requeriments del projecte poden ser viables i quins no a l'hora d'usar l'aplicació en un dispositiu mòbil, per exemple, el format dels arxius de vídeo, actualment pocs mòbils reproduïen formats de "WMV", reproduïen format "MPEG4".
- Si el dispositiu mòbil disposa de GPS incorporat, s'hauria de modificar l'aplicació perquè agafes les coordenades de la imatge del GPS i facilitar l'acció a l'usuari.

11. Conclusions

En l' inici del projecte, la idea de tot el que englobava el projecte era molt clara. Una aplicació web en el que els usuaris poden compartir les seves fotografies i vídeos, amb un suport per poder visualitzar la situació de la imatge en un mapa utilitzant l'API de Google Maps. Abans de començar amb el desenvolupament del projecte he viscut molts contratemps i dificultats, per poder documentar-me respecte la tecnologia més adient per realitzar el projecte.

Vaig triar una tecnologia que pogués interaccionar amb l'API de Google Maps de manera fàcil i precisa, donat que el meu coneixement del funcionament de l'API de Google Maps era pràcticament nul, per tant, vaig haver de documentar-me en el seu funcionament. Finalment, hem vaig decantar per l'opció d'una tecnologia basat en JAVA (JSP) i amb un Framework per complir el patró MVC (Struts) essencial per les aplicacions webs.

Crec que el desenvolupament web, és la tecnologia amb més futur i que agafa cada vegada més pes en la nostra societat. Donat que cada dia, les persones busquen estar molt més informats del que passa en al seu voltant, i l'augment de l'aparició d'Internet en les nostres cases facilita que passi això. Crec que aprendre a desenvolupar aplicacions webs em donarà molta experiència per afrontar projectes futurs, per això vaig escollir aquest projecte.

Ha estat un projecte molt profitós, pel fet que he après a interaccionar amb serveis d'Internet i he ampliat considerablement els meus coneixements de les tecnologies per desenvolupar webs (JSP i Struts) perquè quasi no havia treballat amb elles.

També he après a gestionar un projecte des del començament, analitzant els requeriments i planificant un període de desenvolupament per tal d'arribar a un objectiu final. Tot i haver passat molts contratemps per errades de desenvolupament de codi font, he aconseguit arribar a finalitzar el projecte. Cosa que fa pensar que la meva gestió del projecte ha estat encertada.

Es un projecte interessant per entendre com funcionen les comunicacions entre els serveis d'Internet i com podem treure profit per utilitzar-ho en les nostres aplicacions.

Aquesta documentació es podrà fer servir com a material de consulta per futurs projectistes, que necessitin informació per assolir uns coneixements bàsics sobre les aplicacions web, i poder iniciar els seus projectes.

Bibliografia

1. Títol: CSS, DHTML y Ajax,

Autor: Xavier Colomé Fornós

Editorial: ANAYA

2. JAVASCRIPT

Autor: Tom Negrino , Dori Smith

Editorial: Prentice Hall, Tercera Edición

3. Apunts Laboratori 2

Autor: Joan Jou i Majó

4. JavaScript Edición 2010

Autor: Astor de Caso Parra

Editorial: ANAYA

5. AJAX, JAVASCRIPT Y PHP

Autor: Phil Ballard, Michael Moncur

Editorial: ANAYA

6. <http://www.scribd.com>. Documentació sobre el Framework Struts.

7. <http://www.programacion.com>.

Pàgina web destinada a la programació web i java.

8. <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=fileupload>.

Conceptes sobre el fileupload.

9. <http://www.todoexpertos.com/categorias/tecnologia-e-internet/desarrollo-de-sitios-web>.

Conceptes de la programació web.

10. <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=JakartaCommonsIO>.

Conceptes org.apache.commons.io.FileUtils, per poder afegir arxius a un FTP.

11. http://www.pcweb.es/manual_html/mostrar.php?opcion=Bienvenido_a_tu_manual.

Manual HTML online.

12. <http://lineadecodigo.com/html/cargar-un-video-avi-asf-o-wmv-en-mi-pagina-web>.

Manual HTML, per saber com visualitzar vídeos i fotos en un HTML.

13. <http://www.forosdelweb.com/f15/upload-files-215358>.

Conceptes sobre el UploadFile.

14. <http://www.forosdelweb.com>.

Solució de dubtes respecte la programació web.

I Annex Connectar JAVA-MySQL

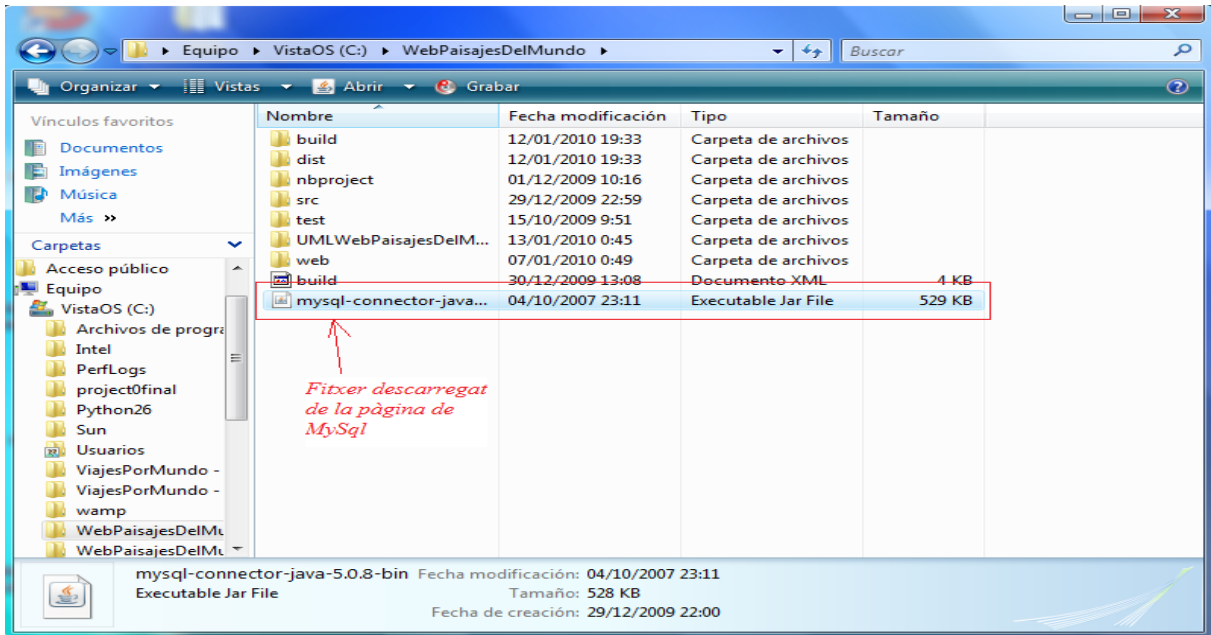
Connectar amb una base de dades MySQL

En aquest apartat es vol fer de guia per realitzar una connexió a una base de dades MySQL. No té com a objectiu ser una guia de referència sobre com fer la connexió MySQL, sinó que busca ajudar i orientar en una de les possibles implementacions d'aquesta connexió.

Per connectar amb una base de dades MySQL el primer que s'ha de fer descarregar el “*mysql-connector-java-5.0.8-bin.jar*”¹¹ de la pàgina oficial de MySQL.

Es pot posar el fitxer en la carpeta de Java que hi ha en el sistema operatiu, però no és una acció molt recomanable, ja que si el desenvolupador canvia de màquina per continuar treballant, la connexió de la base de dades no funcionarà, haurà de modificar l'arxiu de connexió a la BD i afegir un altre vegada el fitxer en la carpeta Java de la nova màquina.

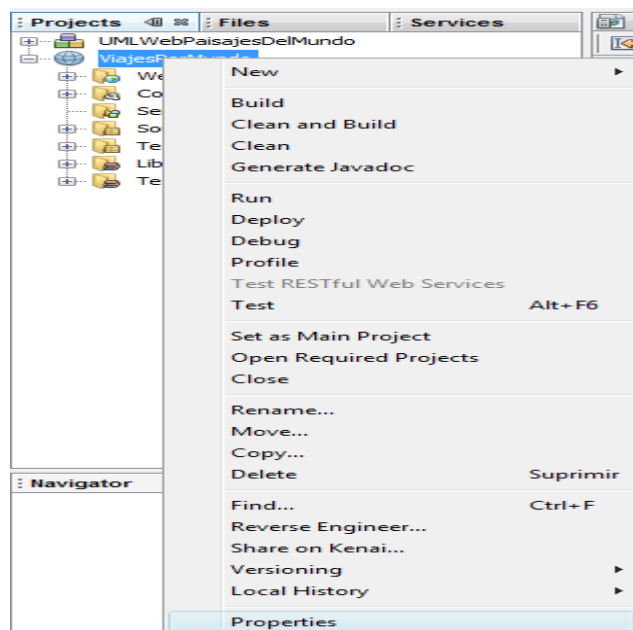
La segona opció es posar el fitxer dintre del mateix projecte, d'aquesta manera si el desenvolupador canvia de màquina per continuar treballant, no tindrà que modificar l'arxiu de connexió a la BD i el fitxer sempre estarà en el projecte *Imatge I Annex 1*.



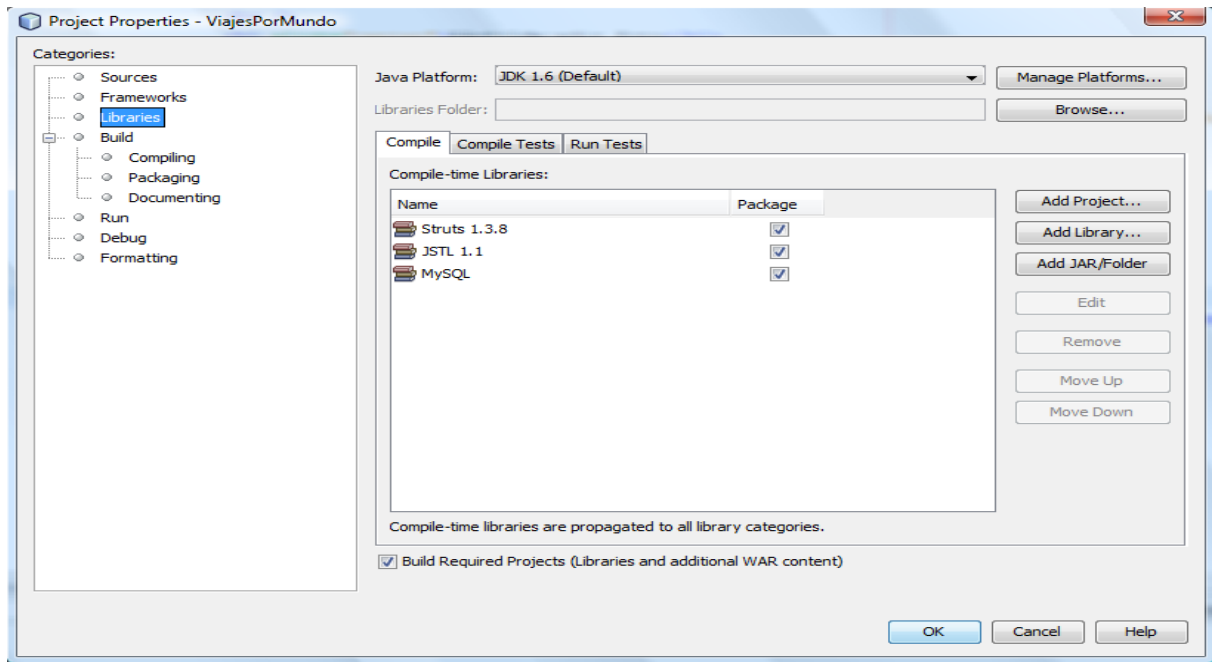
Imatge I Annex 1: Situació del fitxer dintre del projecte

A continuació, s’ha d’afegir la llibreria que fa referència a aquest fitxer perquè el projecte trobi els arxius i drivers suficients per crear dita connexió.

Primer s’ha d’anar a les propietats del projecte *Imatge I Annex 2* . Una vegada situats en la finestra de les propietats s’ha de seleccionar del llistat de les propietats “Libraries”.

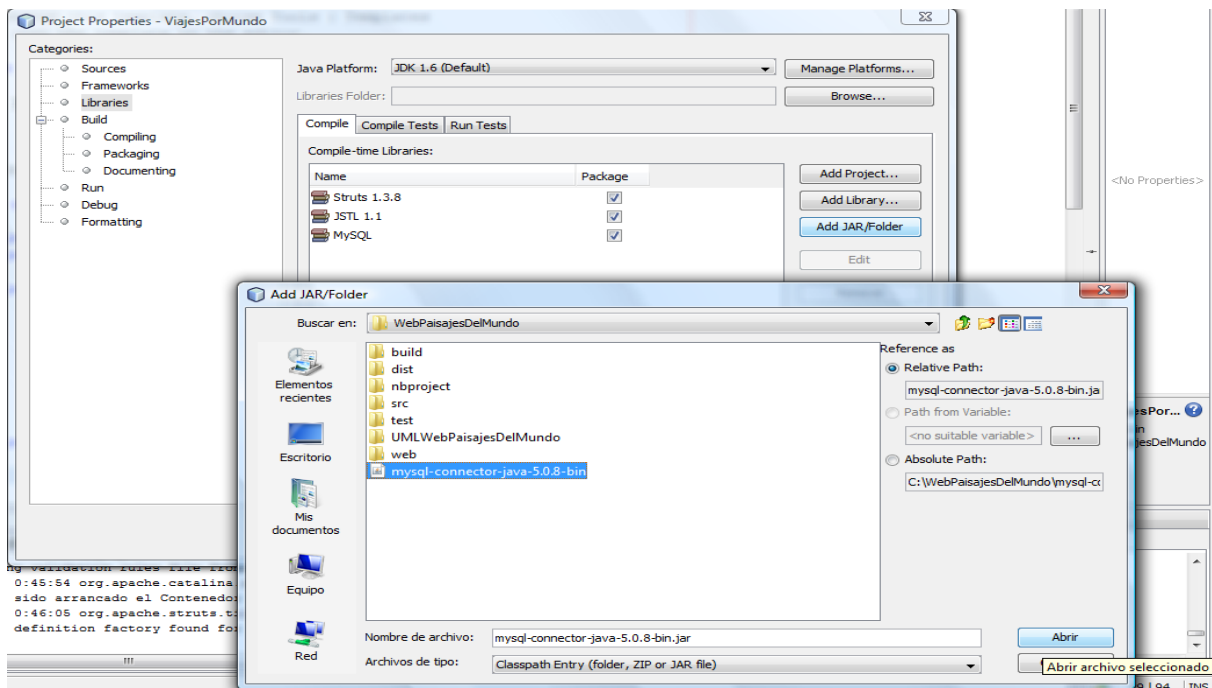


Imatge I Annex 2: Propietats d’un projecte



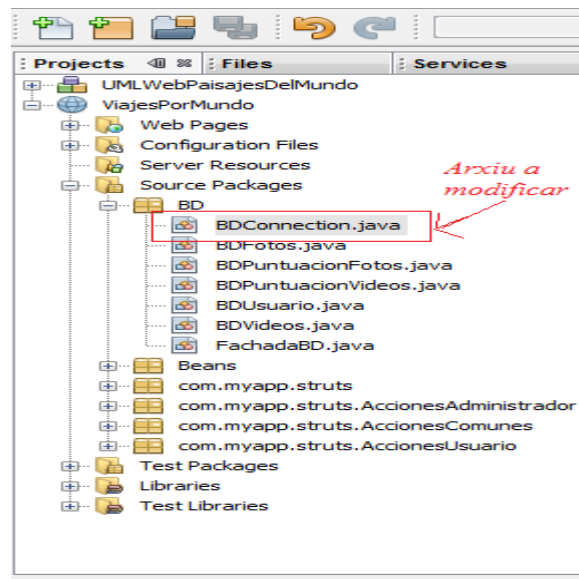
Imatge I Annex 2: Propietats del projecte

S’ha de fer clic en “Add JAR/Folder” i s’obrirà una finestra de diàleg, en la que s’ha de buscar la situació del fitxer “mysql-connector-java-5.0.8-bin.jar” i fer clic en Ok, Imatge I Annex 3.



Imatge I Annex 3: Buscar el fitxer de connexió

Per últim, s'ha de crear un arxiu de BDConnection.java (o modificar si ja s'ha creat per connectar amb una BDAccés) *Imatge I Annex 4*. El primer és modificar el constructor del BDConnection(), en aquest constructor s'ha de carregar el driver de MySQL com s'observa en *Imatge I Annex 5*. La següent acció que s'ha de fer es modificar el mètode que s'ha de connectar a la BD MySQL com s'observa en la *Imatge I Annex 6*.



Imatge I Annex 4: Arxiu BDConnection que s'ha de modificar o crear

```
public class BDConnection {
    private Connection con;
    private static BDConnection bdc;

    private BDConnection() throws Exception {
        try {
            Class.forName("com.mysql.jdbc.Driver");
        } catch (Exception e) {
            throw new Exception("Driver ODBC no trobat, " + e.getMessage());
        }
        this.hacerConexion();
    }
}
```

Carrega del Driver MYSQL

Imatge I Annex 5: Modificar el constructor BDConnection

```
}

private void hacerConexion() throws Exception {
    try {
        con = DriverManager.getConnection("jdbc:mysql://localhost:3306/NOM DE LA BD MYSQL", "USUARI", "CLAU ACCES");
    } catch (Exception e) {
        throw new Exception("Errada en el nom de Connexió: " + e.getMessage());
    }
}
```

Paràmetres de la connexió a la BD

Imatge I Annex 6: Connexió a la BD MySQL

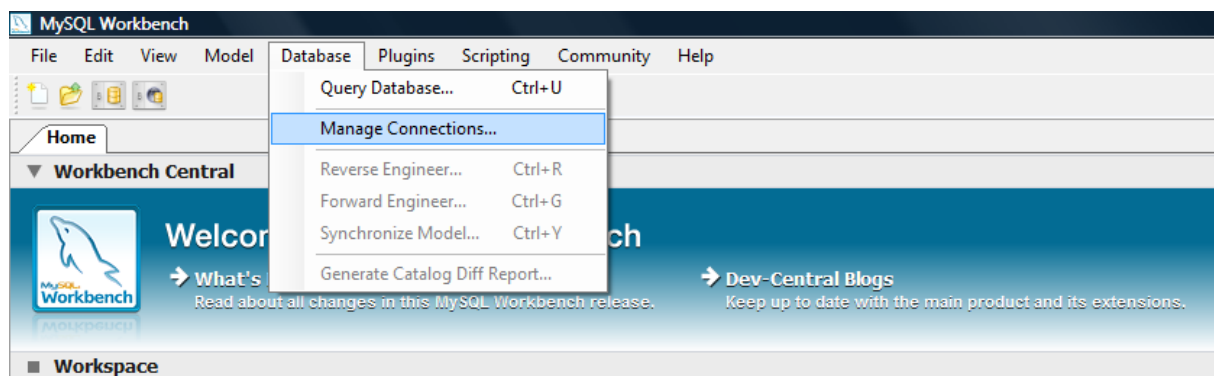
II Annex Workbench 5.2 OSS

Instal·lar Workbench 5.2 OSS

En aquest apartat es vol fer una guia per realitzar una instal·lació d'un programa gestor de base de dades MySQL, aquest programa és el Workbench. Té com objectiu ajudar i orientar en la instal·lació i configuració del programa. I també com realitzar les operacions bàsiques. Aquest programa serveix per modelar les taules de la base de dades com si es treballés amb interfície gràfica (com ho fa el Power Designer), i no per comandes SQL des de Ms-Dos com es fa tradicionalment.

La primera acció a fer es realitzar la descarrega del Workbench 5.2. OSS¹² de la pàgina oficial de MySQL. Després d'haver descarregat el fitxer fem doble clic per instal·lar el programa amb les opcions per defecte.

Quan s'hagi instal·lat el programa, executem el programa i configurarem la connexió amb la Base de Dades MySQL. Anem al menú superior on posa Database i Manage Connection, *Imatge II Annex 1.*



Imatge II Annex 1

Fem clic per crear una nova connexió amb la base de dades. Introduïm les dades que caracteritzen la nostra Base de Dades.

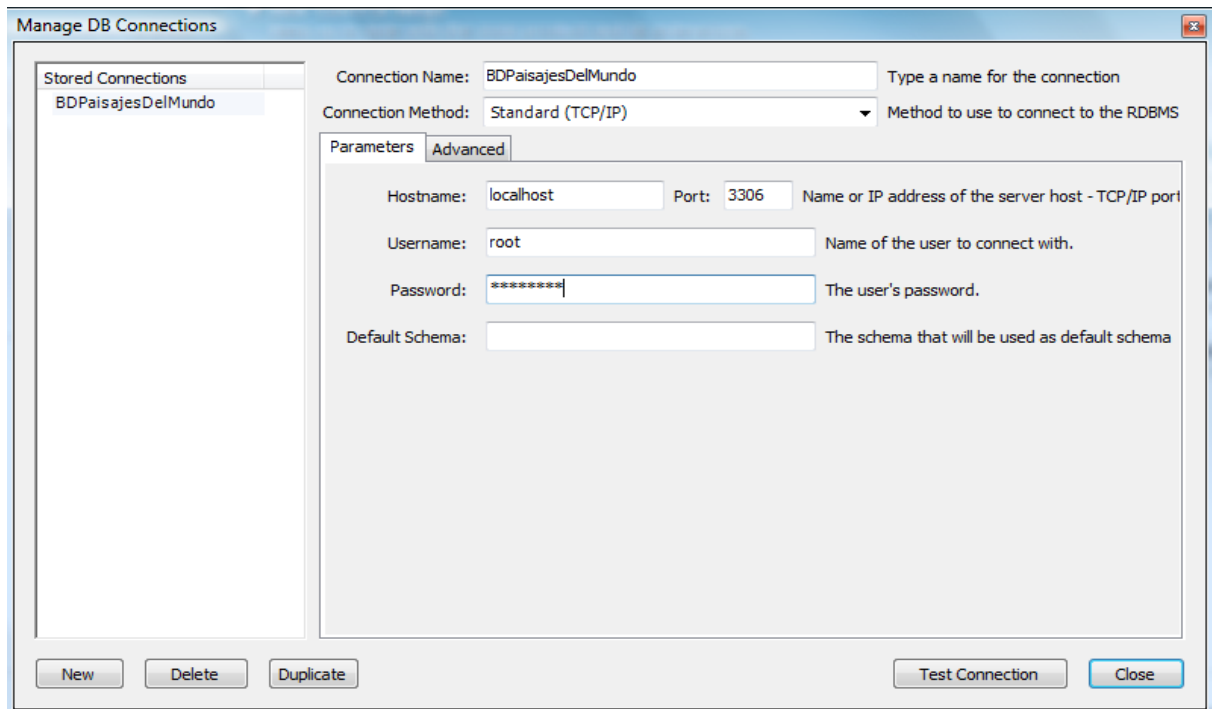
¹² <http://dev.mysql.com/downloads/workbench/5.2.html>

En “Connection Name” posem un nom per identificar la connexió (no importa el nom es pot deixar amb el nom per defecte).

En “Connection Method” posem Standard (TCP/IP).

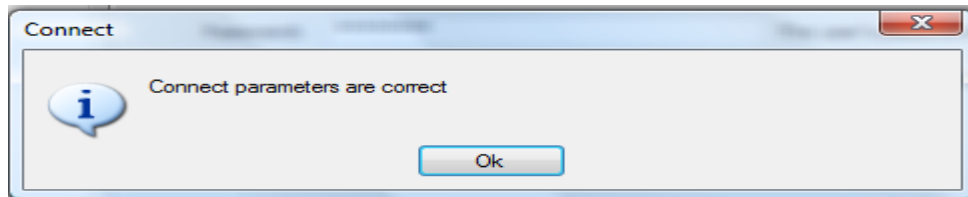
Ara configurarem els paràmetres de connexió, “Hostname” posem localhost i el “Port” posem 3306 si treballem de manera local. En cas contrari posarem l’IP de l’equip servidor i el seu port d’accés.

En el “Username” posem root i en “Password” la clau d’accés a la Base de Dades, *Imatge II Annex 2*. No fa falta realitzar la configuració avançada.



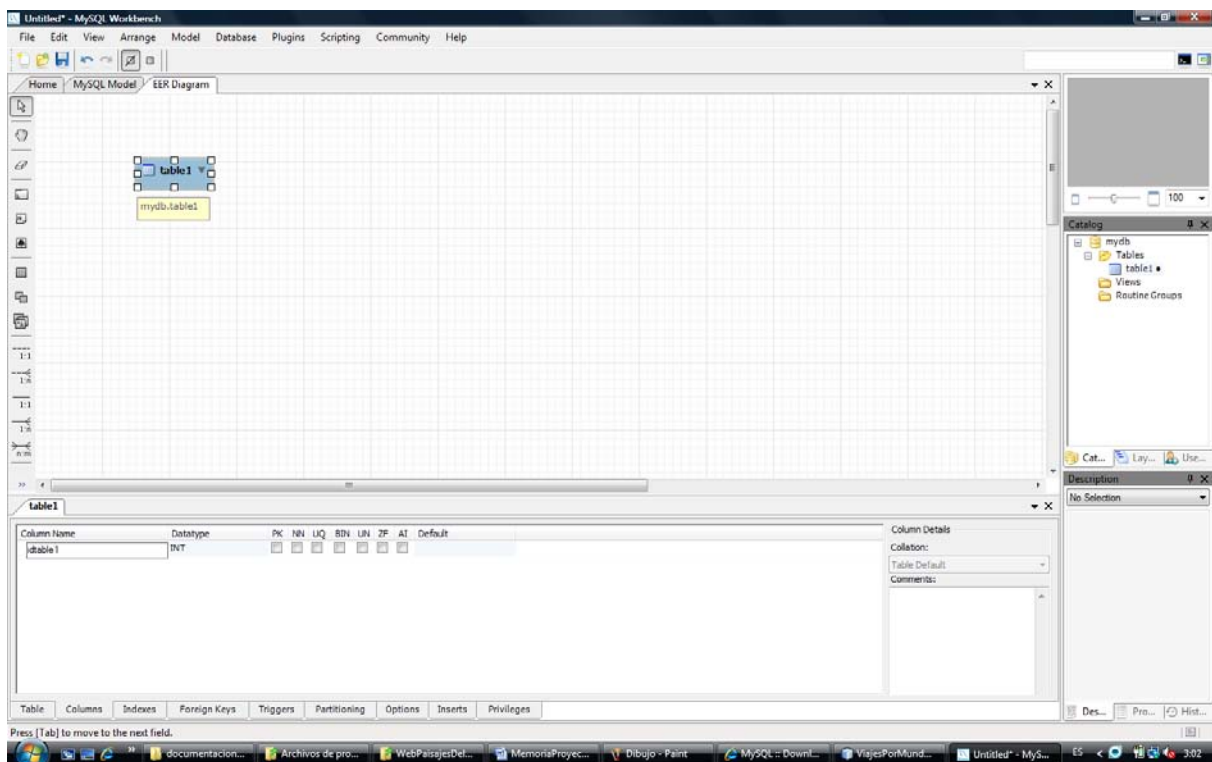
Imatge II Annex 2: Configuració connexió del Workbench amb la base de dades

Per comprovar que s’ha configurat correctament la connexió es fa clic en “Test Connection”, si la connexió es realitza correctament mostrarà el següent missatge:



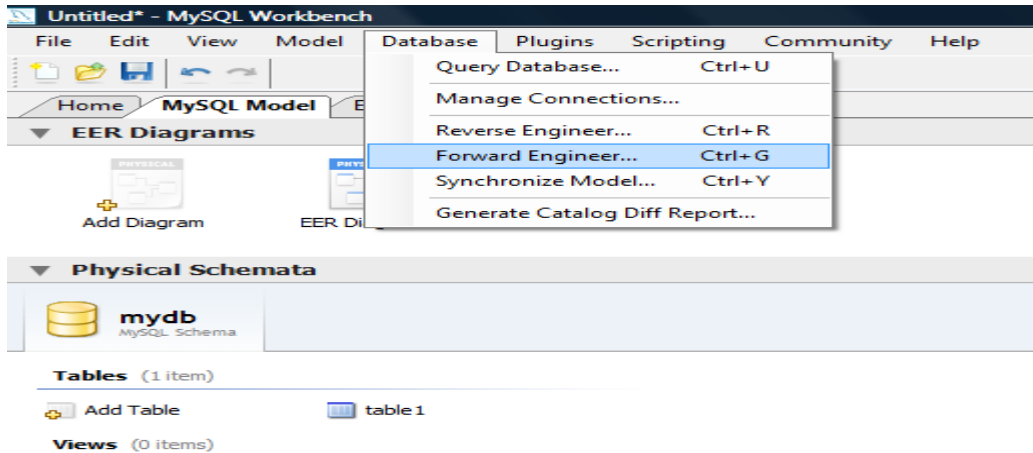
Modelar una nova BD

Fem clic en “Create new EER Model” sortirà la següent pantalla *Imatge II Annex 3*, on podrem crear un diagrama de EER, crear una taula, posar els seus atributs a aquestes taules, crear associacions entre les taules, etc...

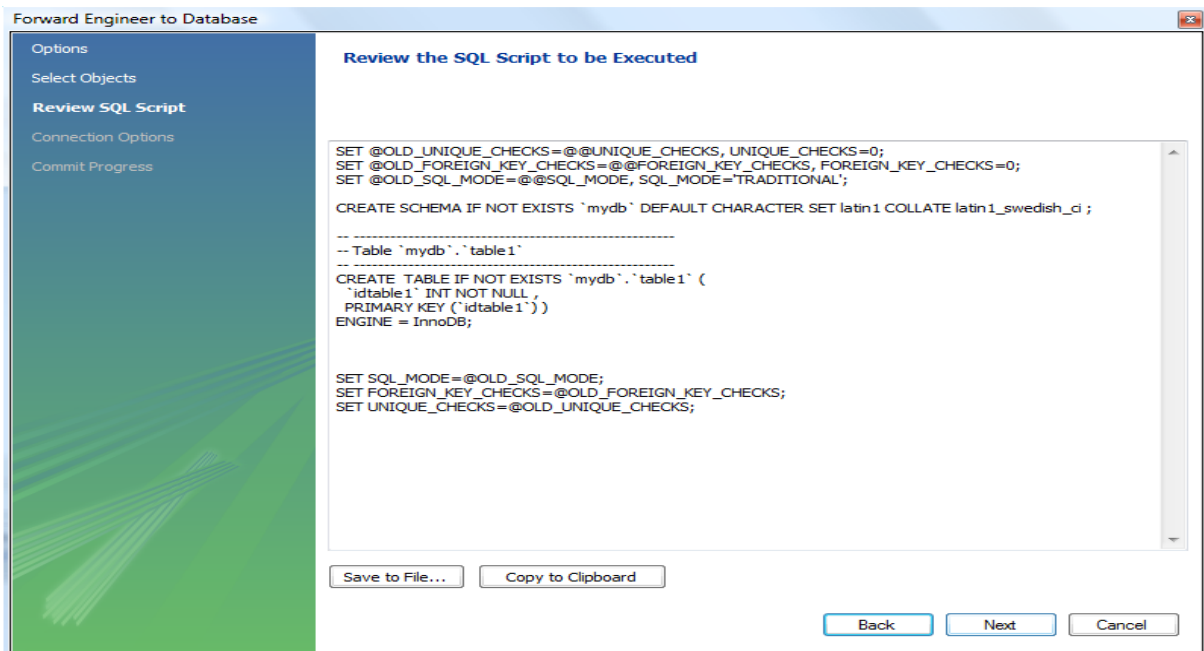


Imatge II Annex 3: Pantalla creació i modelatge de taules en el Workbench

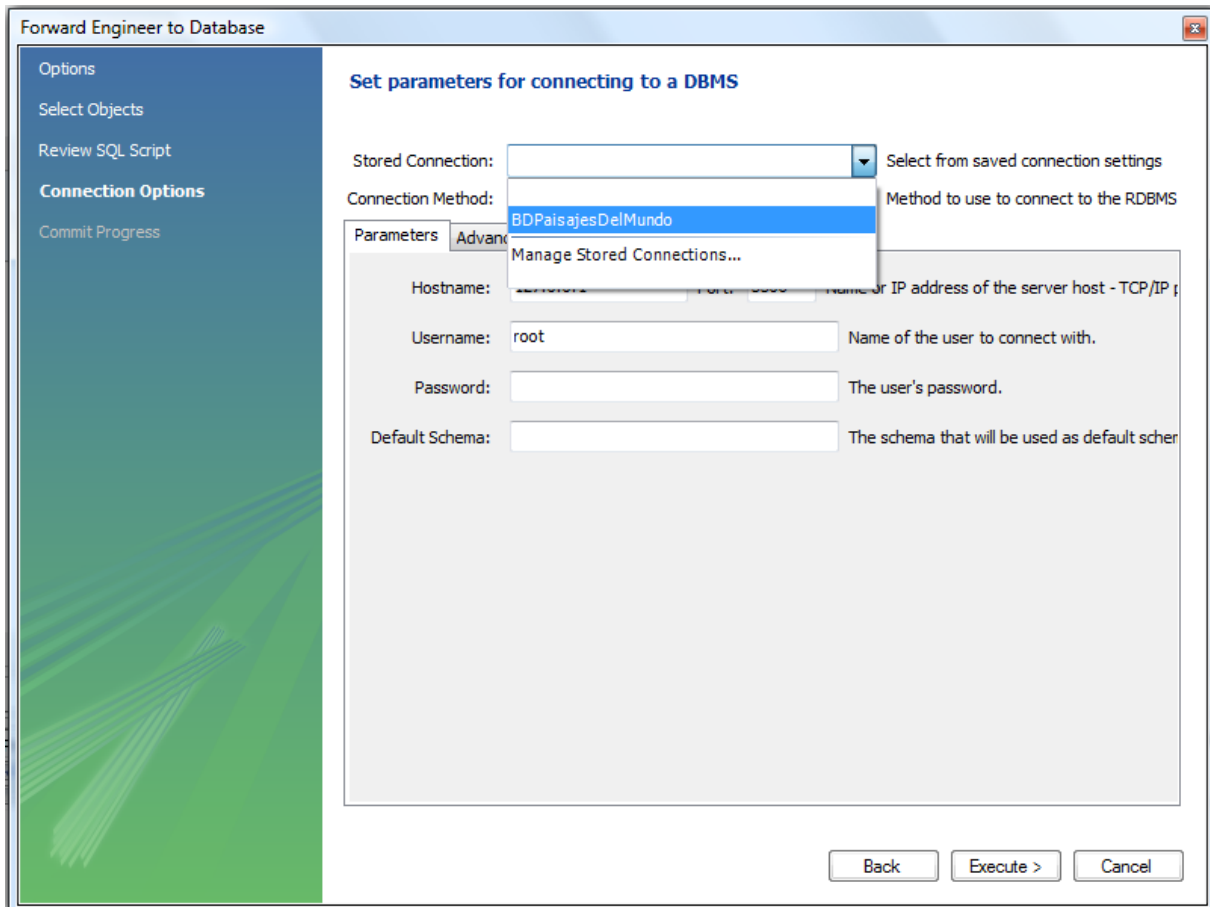
Una vegada creades totes les taules que s’ha considerat necessàries, es pot fer que aquestes taules creades passin a formar part de la base de dades MySQL fent un “Forward Engineer”.



Sortirà un diàleg en el que es pot seleccionar les opcions per fer un “Forward Engineer”. Es fa clic en “Next”. Sortirà un nou quadre de diàleg que demana la confirmació dels objectes que volem afegir a la base de dades MySQL. Es fa clic en Next. Mostrà el script que s’executarà per modificar la base de dades MySQL:



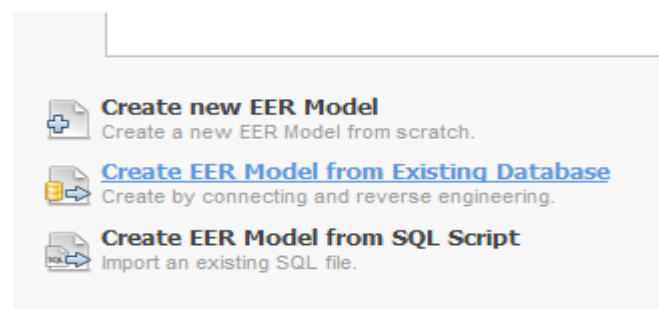
Quan donem a “Next” demanarà la connexió que s’ha configurat anteriorment. La seleccionem i es fa clic en “Execute”, s’executarà l’script i si tot a sortit bé no donarà cap error *Imatge II Annex 4*.



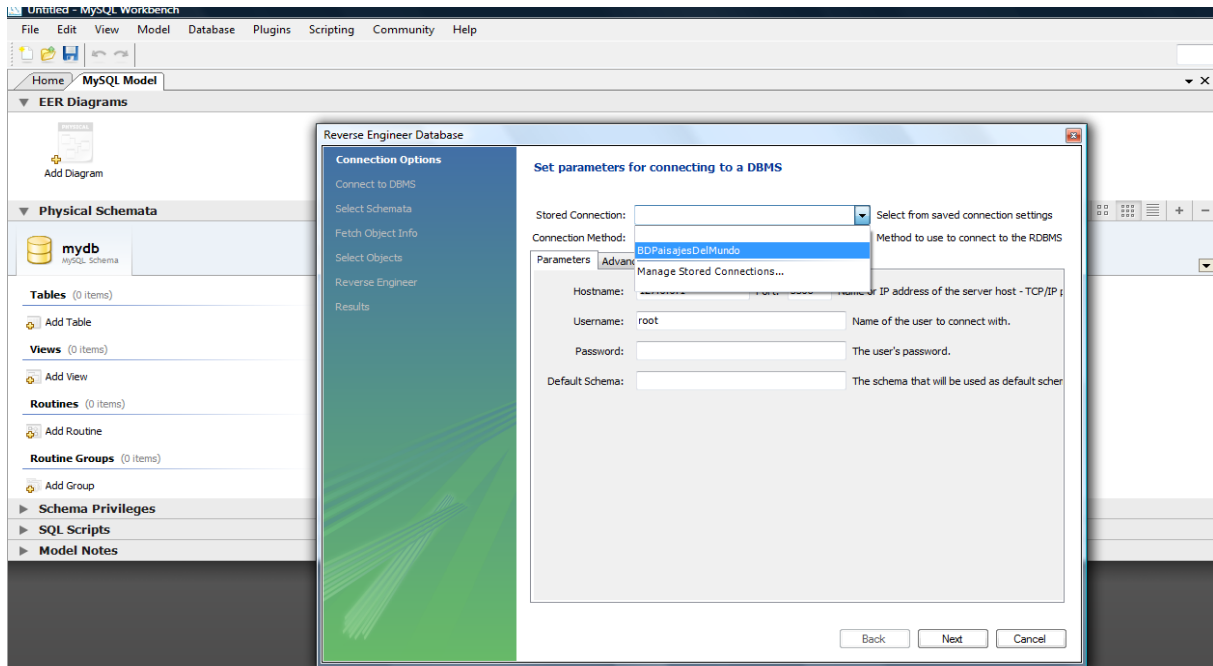
Imatge II Annex 4: Selecció de la connexió creada anteriorment

Modelar una BD ja existent

Si per cas contrari es vol modificar una BD ja existent, es fa clic en “Create EER Model from Existing Database”, *Imatge II Annex 5*. Sortirà un diàleg en el que s’ha d’escollir la connexió que creada anteriorment per tal que el programa pugui fer el “Reverse Engineer”, *Imatge II Annex 6*.

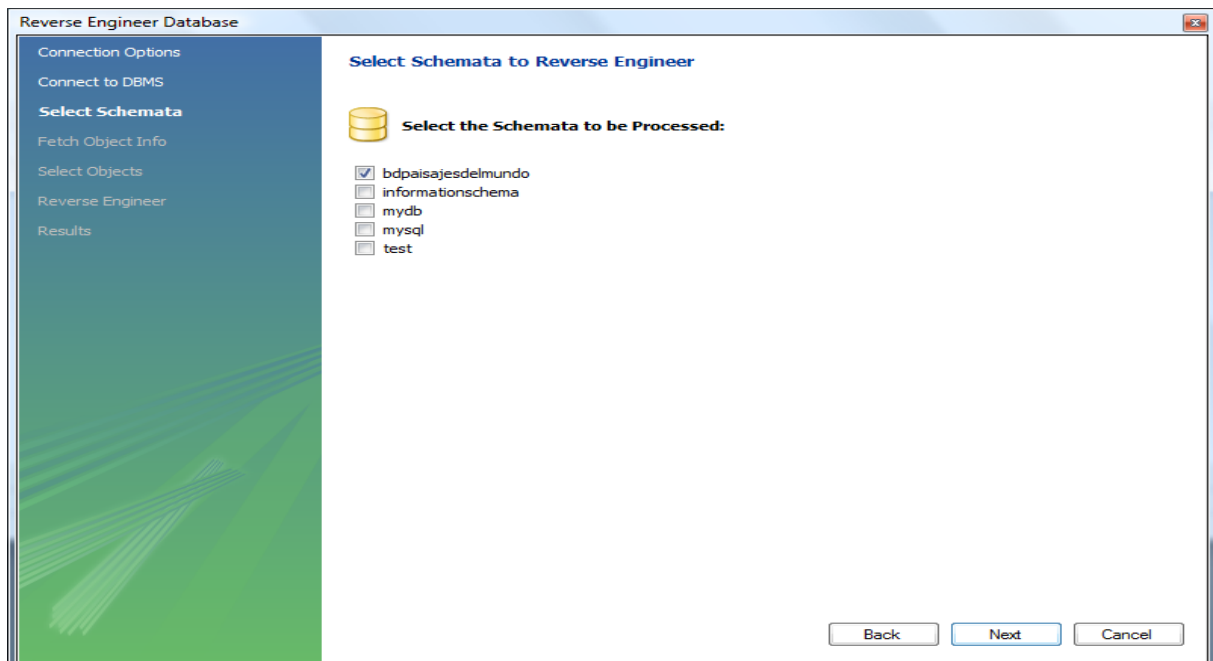


Imatge II Annex 5



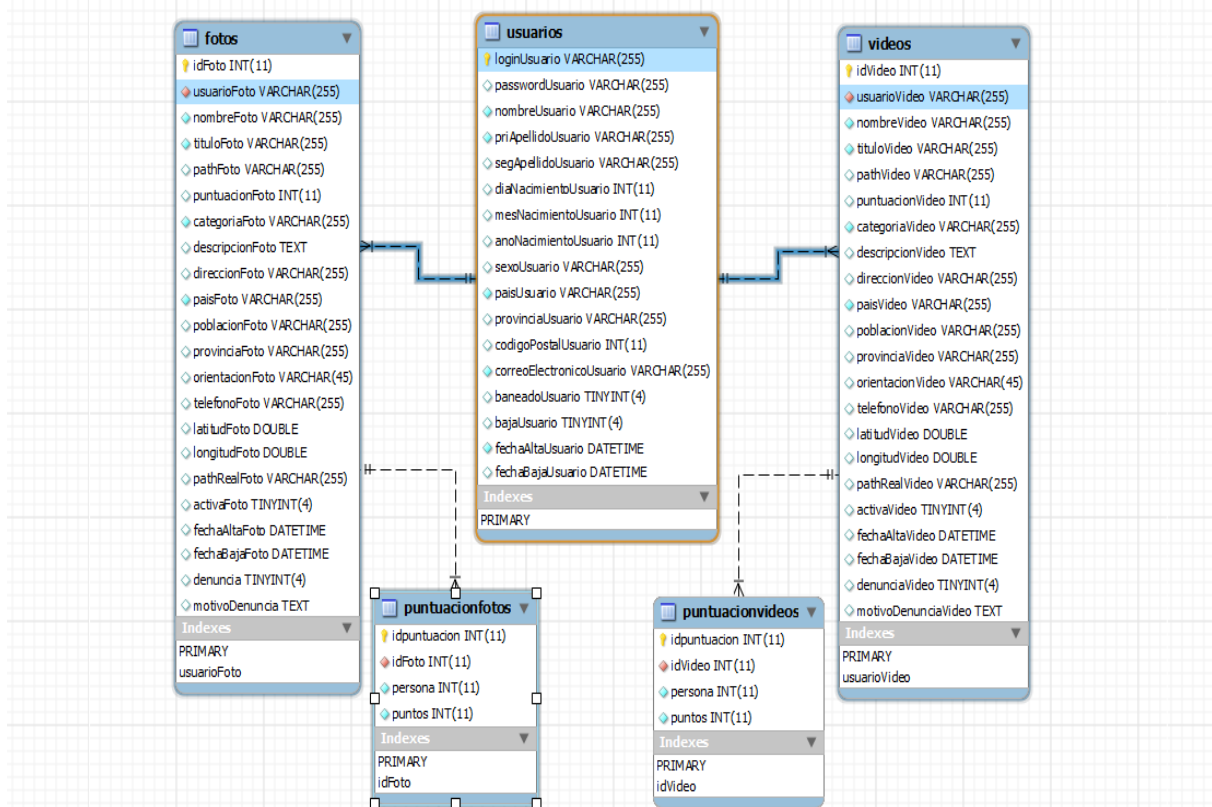
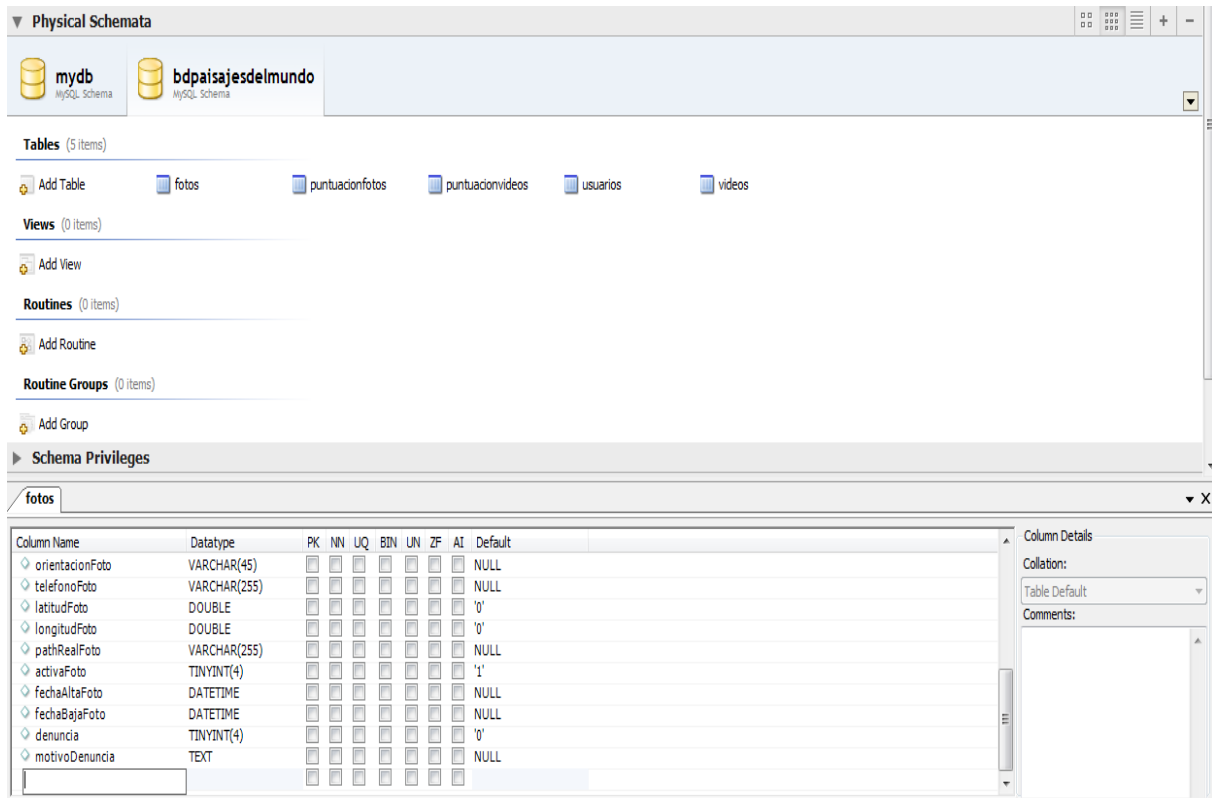
Imatge II Annex 6: Selecció de la connexió a la base de Dades

Després d’haver escollit la connexió es fa clic en “Next”. S’obrirà un quadre de diàleg que demana la selecció de la base de dades que es vol modificar, es selecciona i es fa clic en “Next”, *Imatge II Annex 7*.



Imatge II Annex 7: Selecció BD

Com en el cas del “Forward Engineer” el quadre de diàleg demana la confirmació dels objectes que es vol recuperar, simplement es fa clic en “Execute”. I ja tenim la base de dades per poder modelar i canviar tots els camps de les taules, els índexs, les claus primàries, etc...

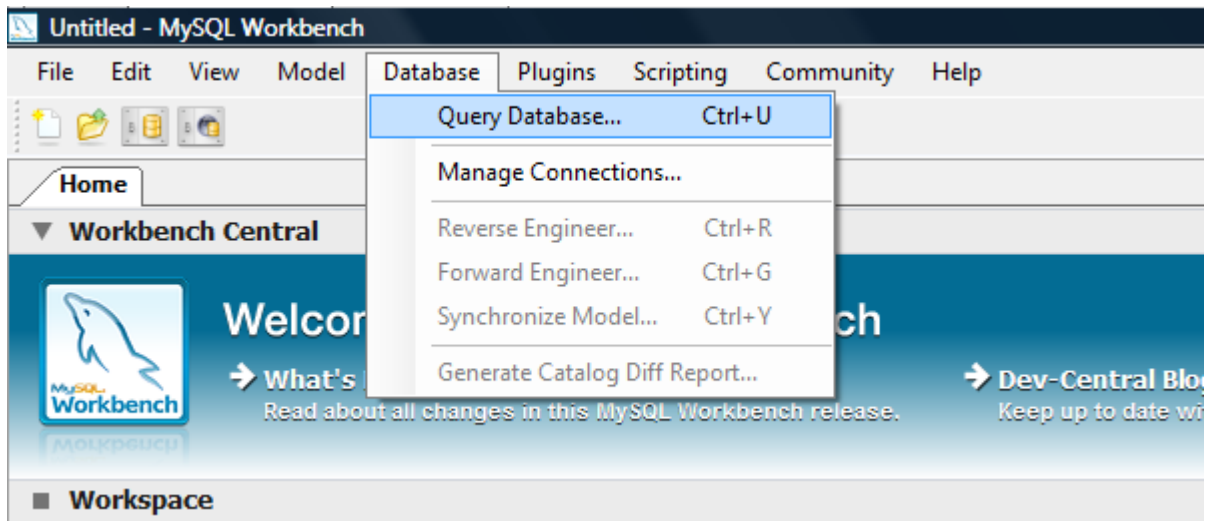


Introduir dades dins d'una taula

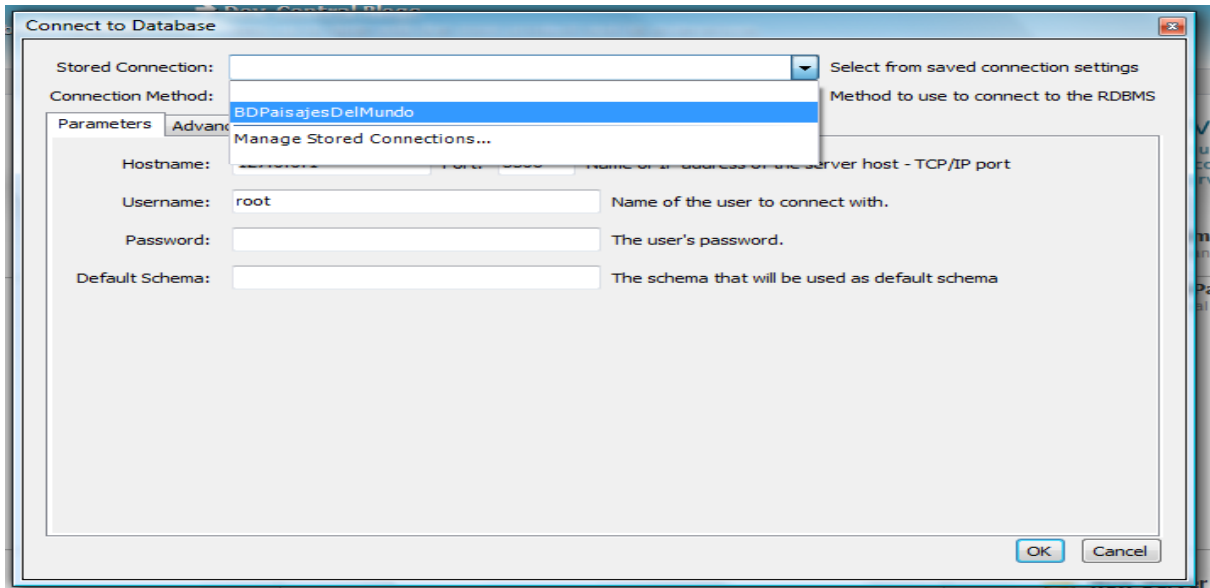
Si no s'utilitza aquest programa a l'hora d'introduir dades s'ha d'introduir amb sentències SQL. Això fa que es perdi molt de temps en introduir-les, i a més, s'ha de vigilar en no tenir cap error de "Sintaxis" en la sentència, perquè llença un error i és costós trobar on s'ha produït l'errata.

El gran avantatge és que, amb aquest programa la introducció de dades es fa amb una interfície gràfica (similar a la de Microsoft Access). L'únic que s'ha de vigilar de fer un "Commit" després de cada introducció de dades.

Per introduir dades en una taula dades s'ha d'anar al menú superior i fer clic en "Database" i "Query Database".



S'obra un quadre de diàleg, en aquest quadre el programa fa la demanda de la connexió a la base de dades que s'ha creat anteriorment.



Es selecciona la connexió i es fa clic en “OK”, automàticament el programa obrirà la base de dades. Només s’ha de seleccionar la taula en la que es vol introduir dades, fer clic en ella i s’obrirà l’editor. En l’editor (*Imatge II Annex 8*) es pot anar introduint dades en les files com si fos l’editor de l’Access (*Imatge II Annex 9*), tenen el mateix comportament i característiques similars.

SQL Statements

```
1 EDIT `bdpaisajesdelmundo`.`usuarios`
```

Overview Output History usuarios (1)

Export

	loginUsuario	passwordUsuario	nombreUsuario	priApellidoUsuario	segApellidoUsuario	diaNacimientoUsuario	mesNacimientoUsuario
	administrador	Nv98b2sA	administrador	administrador	NULL	22	1
	David	1234	David	Luna	Díaz	15	1
	Esther	1234	Esther	Alvarez	Cordero	2	11
	Sergio	1234	Sergio	Luna	Díaz	18	2
	Pepe	123456	PEP	NULL	NULL	NULL	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

*Introducció de
noves dades*

Imatge II Annex 8: Editor per introduir dades Workbench 5.2 OSS

loginUsuario	passwordUs	nombreUsu	priApellidoUsuario	segApellidoUsuario	diaNacimie	mesNacimie	anoNacimie	sexo
a	12	a	l		12	9	2012	Hom
administrador	Nv98b2sA	administrador	administrador		25	6	1984	
Aitor	1234	Aitor	a		13	2	2003	Hom
Antonio	1234	Antonio	l		1	1	2009	Hom
David	1234	David	Luna		1	1	2009	hom
Esther	1234	Esther	l		17	12	1979	Muje
Joan	1234	Joan	Jou		29	11	1968	Hom
Jona	1234	Jona	j		16	6	2000	Hom
Jonas	1234	Jonas	j		9	4	1986	Hom
Manolo	1234	Manolo	m		11	2	1970	Muje
Maria	1234	Maria	m		12	5	1964	Muje
Pepe	12	Pepe	p		1	1	1983	Hom
Sara	1234	Sara	s		13	11	1981	Hom
Sergio	1234	Sergio	L	Díaz	22	1	1984	Hom
Xavi	1234	xavi	l		14	9	1979	Hom
Pepe								

Imatge II Annex 9: Editor per introduir dades Access 2007

Molt important. Quan s'ha acabat d'introduir les dades desitjades en una taula s'ha de fer una última acció, sinó tots els canvis realitzats en les dades de la taula no quedaran guardades. L'acció a realitzar és un "Commit", que es troba situat en la part inferior dreta de l'editor i d'aquesta manera les dades si quedaran guardades.

