



# Aplicació web per a la gestió de continguts multimèdia de l'àmbit ferroviari

**Juan Manuel Alcina Roda**  
Enginyeria Tècnica de Telecomunicació

**Consultor: Manel Llopart Vidal**

8 de gener de 2014

## FITXA DEL TREBALL FINAL

<b>Títol del treball:</b>	<i>APLICACIÓ WEB PER A LA GESTIÓ DE CONTINGUTS MULTIMEDIA DE L'ÀMBIT FERROVIARI.</i>
<b>Nom de l'autor:</b>	<i>JUAN MANUEL ALCINA RODA</i>
<b>Nom del consultor:</b>	<i>MANEL LLOPART VIDAL</i>
<b>Data de lliurament (mm/aaaa):</b>	<i>01/2014</i>
<b>Àrea del Treball Final:</b>	<i>APLICACIONS MULTIMEDIA DE NOVA GENERACIÓ</i>
<b>Titulació:</b>	<i>ENGINYERIA                    TECNICA                    DE TELECOMUNICACIÓ</i>

### **Resum del Treball:**

Aquest treball descriu els passos seguits per desenvolupar una aplicació Web que permeti gestionar continguts multimèdia de l'àmbit ferroviari.

L'aplicació Web desenvolupada, accessible de forma remota mitjançant un navegador Web, permet la visualització de vídeos, imatges i documents que es classifiquen en funció de l'estació o trajecte en el que es troben i del tipus d'instal·lació ferroviària a la que fan referència, de manera que els continguts s'emmagatzemen de forma centralitzada, evitant la dispersió dels mateixos.

A més de la visualització dels diferents continguts, l'aplicació permet la gestió, tant dels usuaris que poden accedir a ella, com de les instal·lacions, localitzacions i dels propis continguts.

Per dur a terme la implementació ha estat necessari l'estudi de les diferents tecnologies i llenguatges de programació existents en l'àmbit de les aplicacions Web, aprofundint en les utilitzades en el desenvolupament final, com HTML, CSS, JavaScript o PHP.

Una altra part important ha seguit el disseny i ús de la base de dades, implementada mitjançant MySQL, i la configuració d'un servidor Web Apache.

**Resumen:**

Este trabajo describe los pasos seguidos para desarrollar una aplicación Web que permita gestionar contenidos multimedia del ámbito ferroviario.

La aplicación Web desarrollada, accesible de forma remota mediante un navegador Web, permite la visualización de videos, imágenes y documentos que se clasifican en función de la estación o trayecto en que se encuentran y del tipo de instalación ferroviaria a la que hacen referencia, de forma que los contenidos se almacenan de forma centralizada, evitando la dispersión de los mismos.

Además de la visualización de los distintos contenidos, la aplicación permite la gestión tanto de los usuarios que pueden acceder a ella, como de las instalaciones, localizaciones y de los propios contenidos.

Para llevar a cabo su implementación ha sido necesario el estudio de las diferentes tecnologías y lenguajes de programación existentes en el ámbito de las aplicaciones Web, profundizando en las utilizadas en el desarrollo final como HTML, CSS, JavaScript o PHP.

Otra parte importante ha sido el diseño y uso de la base de datos, implementada mediante MySQL, y la configuración de un servidor Web Apache.

**Abstract:**

This project describes the steps followed to develop a Web application that allows managing multimedia content related to the railway field.

The Web application once developed , accessible remotely via a web browser, allows the viewing of videos, images and documents classified according to the station or journey between stations where they are and the type of railway facilities to which they refer , so that the contents are stored centrally, preventing dispersion of them .

Besides displaying different contents, the application allows the management of both the users that can access it, as well as the facilities, locations and content themselves.

To carry out the implementation it has been necessary to study the different technologies and existing programming languages in the field of Web applications, deepening those used in the final development like HTML, CSS, JavaScript or PHP.

Another important part is the design and use of database implemented with MySQL and the setting up of an Apache Web server.

**Paraules clau (entre 4 i 8):**

Aplicació Web, programació, bases de dades, servidor Web, multimèdia, instal·lacions ferroviàries.

## Resum

Aquest treball de fi de carrera consisteix en el desenvolupament d'una aplicació web per a la gestió de continguts multimèdia de l'àmbit ferroviari, i per tant, el propi títol, deixa entreveure les parts en les quals es centra el projecte.

El desenvolupament de l'aplicació permet l'accés als continguts de forma organitzada al estar centralitzats, classificats i relacionats i a més, al estar enfocada al manteniment de las infraestructures ferroviàries, es poden fer comparatives sobre els diferents elements entre diferents dates per veure la evolució de les instal·lacions i consultar informació que pugui servir de base per a la redacció de projectes o informes entre altres.

La xarxa ferroviària es complexa i esta formada per diferents tipus d'instal·lacions que permeten la circulació de trens amb garanties de seguretat. Aquestes instal·lacions es poden classificar en funció de la seves característiques tecnològiques com poden ser la infraestructura i via, electrificació, instal·lacions de senyalització o telecomunicacions. Aquestes instal·lacions es troben localitzades geogràficament en estacions i trajectes que les uneixen formant diferents línies que en el seu conjunt formen tota una xarxa ferroviària.

L'aplicació Web desenvolupada permet l'accés remot a través d'un navegador Web de diferents continguts com vídeos, imatges o documents que fan referència tant a un tipus d'instal·lació o element ferroviari, com a una localització, i es accessible per diferents tipus d'usuari en funció del seu rol o paper, sent necessari que aquest estigui validat mitjançant usuari i contrasenya per poder veure qualsevol tipus de contingut.

A més de la reproducció de vídeos i la visualització de fotos i documents que és la seva funcionalitat principal, hi ha altres usuaris que també poden realitzar les tasques de gestió de l'aplicació. Aquestes funcionalitats corresponen a l'alta, baixa i modificació de tota la informació, és a dir, localitzacions (estacions i trajectes), instal·lacions (especialitats i elements) i qualsevol dels tipus de continguts. Finalment els administradors podran a més gestionar els usuaris i els seus permisos.

En tota aplicació web, normalment intervenen diferents components com un servidor on resideix la lògica de l'aplicació, o un sistema gestor de base de dades per emmagatzemar la informació i es fan servir diferents tecnologies, aplicacions i llenguatges per tal de poder presentar convenientment la informació a qualsevol usuari que accedeix a l'aplicació. En el treball és fa un repàs dels components que intervenen en les aplicacions Web i quines son les possibles tècniques, així com els llenguatges de programació existents, tractant de forma especial els continguts multimèdia i la transmissió de vídeo. Fruit d'aquesta cerca s'ha fet una tria per escollir els que més es

poden adaptar al projecte, seleccionant Apache com a servidor Web, MySQL com a SGBD, PHP com a llenguatge del costat del servidor i HTML, CSS i JavaScript com a llenguatges de presentació i del costat del client.

En el seu disseny s'ha seguit una arquitectura de 3 nivells o capes, presentació, aplicació i dades. La capa de presentació és l'encarregada de la interacció amb l'usuari, la capa d'aplicació és on es troba la lògica del negoci que s'executa en el servidor Web, de manera que rep les peticions dels usuaris mitjançant la capa de presentació on es representen finalment les dades. La navegació a través de ella i la entrada de dades per part d'un usuari afecten a l'estat de la lògica de l'aplicació. Finalment la capa de dades s'encarrega de donar curs a les peticions de dades accedint als repositoris que en el nostre cas serà una base de dades. En aquesta fase s'han definit els casos d'ús de l'aplicació, detallant-ne cadascuna de les funcionalitats, el disseny conceptual i lògic de la base de dades i la interfície de l'usuari.

En la implementació s'ha dut a terme la configuració e instal·lació del programari, el disseny físic de la base de dades, i el desenvolupament de la lògica i la presentació amb la escriptura i documentació dels diferents fitxers amb els llenguatges de programació escollits, destacant l'ús d'algunes de les especificacions de HTML5 i de CSS3 pel que fa a la presentació.

## **Agraïments**

M'agradaria dedicar unes paraules d'agraïment a la meva dona Eva i els meus fills Guillem i Iris per el temps que els he robat amb la dedicació a aquest projecte en concret i a la carrera durant aquestos anys.

Un especial record a la nostra estimada Sonia per la lliçó que ens va donar.

# Índex

1. Introducció.....	12
1.1 Context i justificació del Treball .....	12
1.2 Objectius del Treball.....	12
1.3 Enfocament i mètode seguit .....	13
1.4 Planificació del Treball.....	13
1.5 Sumari de productes obtinguts .....	14
1.6 Altres capítols de la memòria .....	14
2. Estat de l'art .....	15
2.1. Aplicacions Web.....	15
2.1.1. Característiques i arquitectura .....	15
2.1.2. Orígens, evolució i tendències. ....	16
2.2. Components, llenguatges i tecnologies de les aplicacions Web. ....	19
2.2.1. Components.....	19
2.2.2. Llenguatges i tecnologies.....	20
2.2.2.1. Llenguatges del costat del client.....	20
2.2.2.2. Llenguatges del costat del servidor .....	21
2.2.2.3 Tecnologies RIA. ....	22
2.3. Els continguts multimèdia i la transmissió de vídeo.....	25
2.3.1. Formats i contenidors multimèdia. ....	26
2.3.1.1. Formats de àudio.....	26
2.3.1.2. Formats de vídeo.....	27
2.3.1.3. Els contenidors multimèdia.....	27
2.3.2. Transmissió de vídeo. ....	29
2.3.2.1. Generació del corrent de vídeo.....	29
2.3.2.2. Reproductors del corrent de vídeo.....	29
2.3.2.3. Serveis de vídeo sota demanda. ....	29
2.3.2.4. Reproducció de vídeo amb HTML5 .....	30
2.4. Llenguatges i tecnologies escollides. ....	31
2.4.1. Servidor Web Apache .....	31
2.4.2. SGBD MySQL .....	32
2.4.3. El llenguatge del costat del servidor, PHP. ....	32
2.4.4. Llenguatges i llibreries del costat del client. ....	33
3. Disseny.....	34
3.1. Requeriments i funcionalitats de l'aplicació Web.....	34
3.2. Casos d'us de l'usuari. ....	35
3.2.1. Diagrama.....	35
3.2.2. Casos d'us de l'aplicació .....	36

3.2.2.1. Actors .....	36
3.2.2.2. Detall de cada cas d'ús.....	37
3.3. Disseny de la base de dades. ....	42
3.3.1. Disseny conceptual. ....	42
3.3.1.1. Diagrama Entitat - Relació.....	43
3.3.2. Disseny lògic. ....	44
3.3.3. Disseny físic. ....	45
3.4. Interfície d'usuari. ....	45
4. Implementació .....	47
4.1. Aplicacions utilitzades. ....	47
4.1.1. Instal·lació i configuració de les aplicacions .....	47
4.2. Formats dels fitxers dels continguts .....	49
4.3. Disseny físic de la base de dades. ....	51
4.4. Implementació de la lògica i la presentació. ....	53
4.4.1. Implementació de la lògica de l'aplicació. ....	53
4.4.1.1. Connexió amb la base de dades. ....	54
4.4.1.2. Navegació.....	54
4.4.1.3. Accés a l'aplicació. ....	55
4.4.1.4. Desconnexió o sortida de l'aplicació.....	56
4.4.1.5. Canvi de contrasenya. ....	56
4.4.1.6. Gestió d'usuaris. ....	57
4.4.1.6.1. Alta d'usuaris.....	57
4.4.1.6.2. Baixa i modificació d'usuaris .....	57
4.4.1.7. Gestió de localitzacions. ....	58
4.4.1.8. Gestió d'instal·lacions. ....	59
4.4.1.9. Gestió de vídeos.....	60
4.4.1.9.1. Llista de vídeos i opcions. ....	60
4.4.1.10. Gestió de fotos.....	63
4.4.1.11. Gestió de documents.....	64
4.4.1.12. Continguts.....	65
4.4.2. Implementació de la presentació de l'aplicació. ....	66
5. Conclusions i línees de futur .....	70
5.1. Conclusions.....	70
5.2. Línees de futur .....	71
Annexos .....	72
Annex 1. Manual d'instal·lació i configuració de l'aplicació.....	72
Bibliografia.....	73



## Índex de figures

Figura 1. Diagrama de Gantt.....	14
Figura 2. Diagrama de comunicació entre navegador i servidor.....	16
Figura 3. Estructura de 3 capes d'una aplicació Web.....	16
Figura 4. Procés de generació de pàgines dinàmiques.....	17
Figura 5. Estructura d'una RIA.....	18
Figura 6. Models de transport de les aplicacions Web, clàssic i amb AJAX.....	23
Figura 7. Contenidors multimèdia.....	28
Figura 8. Evolució del compliment de HTML5 segons els navegadors.....	30
Figura 9. Compatibilitat navegadors i àudio/vídeo de HTML5.....	31
Figura 10. Diagrama de casos d'ús de l'aplicació.....	35
Figura 11. Esquema Entitat - Relació.....	43
Figura 12. Esquema d'interfície d'usuari.....	45
Figura 13. Pàgina inicial de XAMPP.....	48
Figura 14. Tauler de control de XAMPP.....	48
Figura 15. Interfície gràfica de phpMyAdmin.....	51
Figura 16. Vista de la estructura de la taula "usuarios".....	52
Figura 17. Vista de les taules amb la interfície phpMyAdmin.....	52
Figura 18. Esquema de navegació de l'aplicació en funció del tipus d'usuari..	54
Figura 19. Pantalla de login.....	56
Figura 20. Pantalla inicial després de validat.....	56
Figura 21. Vista del formulari de canvi de contrasenya.....	56
Figura 22. Vista de la pantalla d'usuaris amb els 2 panells desplecats.....	57
Figura 23. Vista del formulari per modificar o eliminar un usuari.....	58
Figura 24. Selecció d'una estació per modificar les dades.....	59
Figura 25. Vista dels formularis de gestió de instal·lacions.....	60
Figura 26. Vista del panell esquerre desplegat amb la llista de vídeos.....	61
Figura 27. Detall de la pantalla per visualitzar un vídeo.....	61
Figura 28. Vista de les opcions de filtrat i dels formularis d'alta, i modificació..	62
Figura 29. Detall del llistat de fotos disponibles i seleccionades.....	64
Figura 30. Visualització de les fotografies seleccionades.....	64
Figura 31. Vista dels 2 panells per seleccionar estació i tipus de contingut....	65
Figura 32. Detall del llistat amb el diferents continguts trobats.....	66
Figura 33. Vista de la finestra modal amb confirmació.....	68
Figura 34. Vista de la finestra modal amb un missatge.....	69
Figura 35. Vista del widget Datepicker de jQuery UI.....	69
Figura 36. Importació de la base de dades.....	72

## Índex de taules

Taula 1. Principals fites del projecte.....	13
Taula 2. Casos d'ús de l'aplicació.....	37
Taula 3. Descripció dels camps de la base de dades.....	42
Taula 4. Valors dels diferents camps de la base de dades.....	44
Taula 5. Compatibilitat navegadors i formats HTML5.....	50

## Acrònims

### AJAX

**Asynchronous JavaScript And XML.** És una tècnica utilitzada en les aplicacions Web que utilitza el llenguatge del costat del client JavaScript i XML de forma asíncrona.

### API

**Application Programming Interface.** És un conjunt d'especificacions per la comunicació entre components software.

### ASP

**Active Server Pages.** Tecnologia de Microsoft del tipus del costat del servidor per a pàgines Web generades dinàmicament. A partir de l'any 2002 es va substituir per ASP.NET.

### CGI

**Common Gateway Interface.** Anomenat interfície d'entrada comuna. Van ser les primeres aplicacions que s'executaven en el servidor Web, i que generen contingut dinàmic per a les pàgines Web.

### CRC

**Class Responsibility Collaborator.** Les targetes CRC són una metodologia pel disseny de software orientat a objectes.

### CSS

**Cascading Style Sheets.** És un llenguatge d'estils utilitzat per definir la presentació, el format i la aparença dels documents de marcat, normalment per a les pàgines Web. Actualment està en vigor la tercera revisió de la especificació pel World Wide Web Consortium (W3C), anomenada CSS3.

### DOM

**Document Object Model.** És una API per a documents HTML i XML que proporciona una representació estructural del document, permetent la modificació del contingut o la presentació virtual. Essencialment comunica les pàgines Web amb *scripts* o llenguatges de programació.

### HTML

**HyperText Markup Language.** És el llenguatge de marcat d'hipertext predominant per a la creació de les pàgines Web.

### HTTP

**HyperText Transfer Protocol.** Protocol que s'utilitza per transmetre informació en el World Wide Web. Principalment es fa servir per obtenir pàgines HTML.

### IP

**Internet Protocol.** És un protocol de la capa de xarxa que s'utilitza per comunicar dades a través d'una comunicació de paquets. IP aporta un servei de adreces úniques i globals per la comunicació entre diferents equips.

## **JSP**

Java **S**erver **P**ages. Tecnologia JAVA que permet generar contingut dinàmic.

## **PHP**

Hipertext **P**re-**P**rocessor (des de 1997). Inicialment s'anomenava eines per a pàgines personals, **P**ersonal **H**ome **P**age **T**ools. És un llenguatge del costat del servidor per permet generar contingut dinàmic.

## **RIA**

Rich Internet **A**pplication. Les aplicacions enriquides de Internet, son un tipus d'aplicacions web que permeten incrementar i millorar les opcions i capacitats de les aplicacions Web tradicionals.

## **TCP**

Transmission **C**ontrol **P**rotocol. És un protocol de la capa de transport que s'utilitza per establir connexions entre equips de la mateixa xarxa.

## **UML**

Unified **M**odeling **L**anguage. El llenguatge unificat de modelatge proporciona una notació i un conjunt de diagrames que permeten representar un model o abstracció d'un problema a resoldre.

## **XAMPP**

**X** Qualsevol sistema operatiu, **A**pache, **M**ySQL, **P**HP i **P**erl. Eina de desenvolupament amb diferents aplicacions, programari, llibreries i utilitats per a manegar y administrar una aplicació Web.

## **XHTML**

e**X**tensible **H**yper**T**ext **M**arkup **L**anguage. Es tracta bàsicament de llenguatge HTML expressat com a XML vàlid.

## **XML**

e**X**tensible **M**arkup **L**anguage. Llenguatge de marcat ampliable o extensible. Desenvolupat pel World Wide Web Consortium (W3C).

# 1. Introducció

## 1.1 Context i justificació del Treball

Actualment la meua feina té relació amb el manteniment de les infraestructures de la xarxa ferroviària, i es disposa de moltes fotografies de les instal·lacions, de vídeos dels recorreguts dels trajectes entre estacions, i molta documentació relacionada, com per exemple plànols o esquemes. La possibilitat de consultar aquesta informació és de molta utilitat entre altres coses per tenir informació de l'estat de la infraestructura i dels elements que la formen, per la elaboració de informes, incidències o afeccions per obres en les proximitats de les vies.

En molts casos es disposa d'aquesta informació, però no es troba centralitzada, sinó que esta dispersa en molts casos en diferents ordinadors de diferents usuaris i amb el pas del temps es perd, o no s'aprofita per a la resta d'usuaris.

La proposta de treball pretén desenvolupar una aplicació web per gestionar tots aquests continguts (vídeos, fotos i documents), de forma que estiguin tots centralitzats i accessibles per tots els usuaris en funció del perfil que tingui assignat. D'aquesta forma és pretén evitar la dispersió i pèrdua dels mateixos, disminuir la quantitat de còpies i volum de dades emmagatzemat i aprofitar més els recursos al estar classificats pel tipus d'instal·lació o el lloc on estan situats, i estar disponibles per a la seva consulta per més usuaris. Un altre benefici que es deriva de la implementació de l'aplicació, serà que es podrà veure l'estat dels elements en el pas del temps, de forma que es podran fer comparatives visuals entre diferents dates en punts concrets de la infraestructura ferroviària.

## 1.2 Objectius del Treball

Els objectius del treball són els següents:

- Desenvolupar una aplicació Web per gestionar i mostrar continguts multimèdia que fan referència a les instal·lacions de l'àmbit ferroviari.
- Conèixer i definir el concepte d'aplicacions web, quines són les seves característiques principals i les seves limitacions, la seva evolució i quines tecnologies existeixen actualment.
- Conèixer els diferents elements que intervenen en el funcionament d'una aplicació web (servidors, sistemes gestors de bases de dades), i els llenguatges de programació interpretats pel client i servidor que estan relacionats.

- Conèixer els formats de àudio i vídeo existents, així com dels contenidors multimèdia, com funciona la transmissió de vídeo sota demanda, com es genera el corrent i la recepció, programari disponible i els reproductors.
- Escollir-ne alguns d'ells per tal de portar a terme el disseny i la implementació de l'aplicació Web repassant els coneixements apresos durant la carrera (com per exemple programació i bases de dades), ampliant-los amb nous continguts i aprofundint en el seu coneixement i ús.

### 1.3 Enfocament i mètode seguit

Tal com s'ha explicat en els apartats anteriors, l'objectiu principal és el de desenvolupar una aplicació Web que permeti als diferents usuaris autoritzats poder consultar diferents continguts relacionats amb les instal·lacions ferroviàries. El treball dut a terme, s'ha centrat principalment en desenvolupar un producte nou després de fer prèviament una tasca de recerca sobre els elements, tecnologies i llenguatges que intervenen en el normal funcionament d'una aplicació Web.

Per a la selecció entre les diferents alternatives de desenvolupament, s'ha tingut en compte la voluntat de fer servir en la mesura del possible programari lliure, que en l'ús de l'aplicació no és necessària molta interactivitat amb l'usuari, donat que es preveu uns 50 usuaris com a màxim que difícilment concorreran en el temps, ni cal una interfície especialment "rica" al ser una aplicació que s'utilitzarà en una Intranet, i l'existència d'eines que incorporen aquestes tecnologies i elements que faciliten la seva instal·lació, configuració i gestió de forma molt intuïtiva.

En la part practica ha estat necessari aprofundir en l'ús i programació de llenguatges com PHP (del costat del servidor) o JavaScript (del costat del client), així com de disseny i presentació de pàgines Web, encara que es cert que en Internet es pot trobar molta documentació i exemples que faciliten molt aquesta tasca.

### 1.4 Planificació del Treball

El treball s'ha planificat tenint en compte les fites principals, centrades principalment en els lliuraments de les PACS:

Data	Descripció de la fita
18 de setembre de 2013	Enunciat TFC
2 de octubre de 2013	Lliurament PAC1 Projecte.
23 de octubre de 2013	Lliurament PAC2 Estat de l'art.
15 de desembre de 2013	Lliurament PAC3 Disseny i implementació.
8 de gener de 2014	Lliurament PAC4 Memòria.
17 de gener de 2014	Lliurament PAC5A Presentació.
26 de gener de 2014	PAC5B Defensa.

**Taula 1. Principals fites del projecte.**

El desglossament de les tasques i la seva temporització es resumeix en el següent diagrama de Gantt:

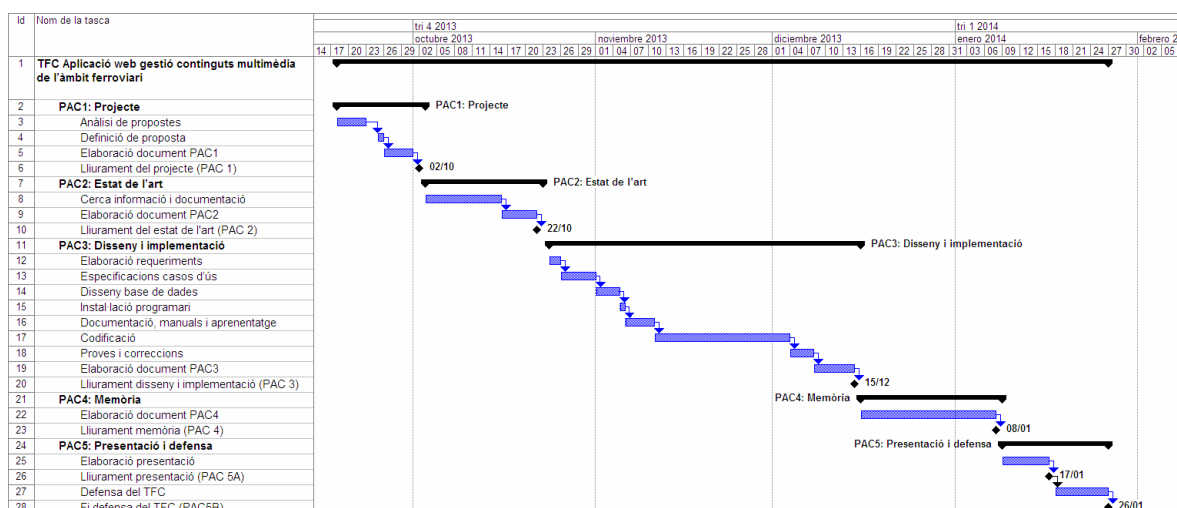


Figura 1. Diagrama de Gantt.

## 1.5 Sumari de productes obtinguts

Els productes obtinguts han estat l'aplicació Web desenvolupada, acompanyat de la present memòria del projecte, presentació (vídeo amb imatge i so on s'expliquen els aspectes més rellevants del treballs realitzat i el resultat obtingut) i codi font.

## 1.6 Altres capítols de la memòria

El capítol 2 descriu la fase de l'estat de l'art, fent una cerca i repàs dels diferents projectes, tècniques i llenguatges que existeixen en la actualitat que estan relacionats amb el TFC a realitzar per tal de conèixer quin és l'estat actual per tal de portar a bon terme la seva realització. Fruit d'aquesta cerca s'escullen els llenguatges, tecnologies i aplicacions que s'utilitzen en les següents fases. En el capítol 3 es descriu el disseny dut a terme en la realització del treball fent servir un diagrama de casos d'ús per mostrar com interactuen i amb quines funcionalitats els diferents actors del nostre sistema i es detalla cadascun dels casos d'ús que defineixen les diferents funcionalitats. També es descriu el disseny conceptual i lògic de la base de dades que servirà de suport a l'aplicació i el disseny general de la interfície d'usuari.

En el capítol 4 es descriu la fase d'implementació, començant per la instal·lació del programari i el disseny físic de la base de dades passant a codificar les diferents funcionalitats. Aquesta implementació es centra en la part lògica de la aplicació implementada fonamentalment amb PHP i la part de presentació, implementada amb HTML, CSS i JavaScript. El capítol 5 tracta sobre algunes de les proves i experiments realitzats i les tasques dutes a terme per comprovar que el funcionament és correcte.

Finalment en el sisè capítol es detallen els objectius assolits, les conclusions i les línies de futur obertes.

## 2. Estat de l'art

El treball, tal com es definia en la etapa de projecte, consisteix en el desenvolupament d'una aplicació web per a la gestió de documents multimèdia de l'àmbit ferroviari, i per tant, el propi títol, deixa entreveure les parts en les quals es centra el projecte.

El objectiu principal és el de desenvolupar una aplicació web, i per tant s'ha de veure quines son les diferents parts, tècniques i llenguatges que intervenen e interactuen normalment, per tal de que posteriorment en la fase de disseny, escollir les opcions que més s'adapten a les necessitats i requeriments del projecte. Aquesta aplicació web evidentment té una funcionalitat definida, que en aquest cas és la gestió de continguts multimèdia de l'àmbit ferroviari. La gestió comporta normalment l'alta, baixa modificació o consulta d'informació, i per tant, serà necessari l'ús d'una base de dades per tal de donar suport a l'aplicació. Cal tenir en compte que es gestionaran continguts com vídeos, imatges o documents i per tant és necessari veure els diferents formats existents i les opcions i tecnologies disponibles per tractar-los i reproduir-los. Finalment els documents a gestionar fan referència a instal·lacions i recorreguts de l'àmbit ferroviari, i per tant, les diferències respecte a la gestió d'altres tipus de continguts, seran simplement a nivell funcional.

### 2.1. Aplicacions Web.

En enginyeria del software s'anomenen aplicacions Web, als programes o eines ubicats en un servidor Web que permeten interacció entre el usuari i l'ordinador, i que ofereixen un serveis determinats als usuaris a través de Internet o d'una Intranet mitjançant un navegador. Existeix molta informació referent als aspectes bàsics de les aplicacions Web, els elements que intervenen, o les tecnologies i llenguatges que s'utilitzen, com per exemple en [1], relacionats fonamentalment amb el programari lliure. En els següents apartats es fa un repàs d'aquestos.

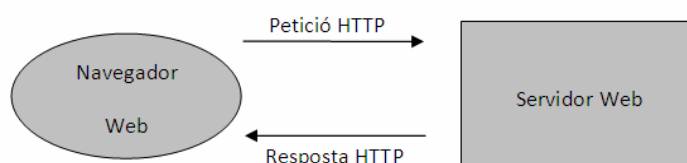
#### 2.1.1. Característiques i arquitectura

Les característiques principals d'una aplicació web son les següents:

- Esta allotjada en un servidor Web i segueixen una arquitectura client – servidor.
- Es accessible mitjançant un navegador Web a través d'una xarxa com Internet o una Intranet.
- Son multiplataforma, es a dir, es poden executar en qualsevol sistema operatiu que tingui un navegador Web.
- S'executa integrament en el servidor Web i pot interactuar amb altres processos i aplicacions.

- La navegació a través de l'aplicació i la introducció de dades per part d'un usuari afecten a l'estat de la lògica de l'aplicació.
- La lògica de la aplicació s'executa en el servidor Web, mentre que el navegador del client només representa les dades.

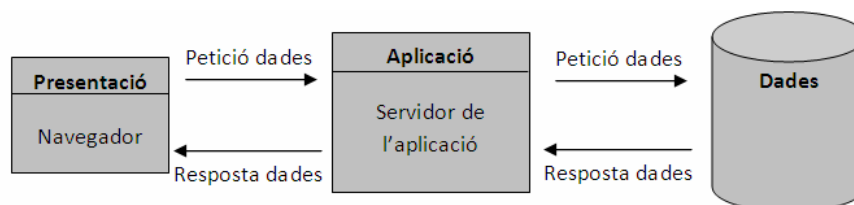
L'arquitectura client – servidor és un model d'aplicació distribuïda en la que les tasques es reparteixen entre els proveïdors de serveis o recursos (servidor Web en el nostre cas) i els demandant que és el client (navegador). Associat a l'arquitectura, apareix el concepte de protocol, que es coneix com el llenguatge utilitzat per comunicar-se entre els dos elements de la xarxa, i que en el cas de les transaccions de la Web, s'anomena HTTP.



**Figura 2. Diagrama de comunicació entre navegador i servidor.**

D'altra banda les aplicacions Web estan estructurades en capes o nivells funcionals independents, i que normalment son 3, la capa de presentació, la capa d'aplicació (o lògica) i la capa de dades.

La capa de presentació és l'encarregada de interaccionar amb l'usuari, de manera que envia les peticions o consultes generades per l'usuari, rep les dades enviades pel servidor i les representa en el navegador. En la capa d'aplicació és on es localitza la lògica del negoci i les seves funcions principals son les de obtenir e interpretar les peticions de la capa de presentació i enviar-li les dades de resposta, executar la lògica necessària i la obtenció i petició de dades cap als repositoris d'informació (normalment bases de dades). Finalment la capa de dades s'encarrega de rebre les peticions de dades cridant a les funcions d'accés a les dades, realitza el procediment o càlcul demanat i torna la informació cap a la capa d'aplicació.



**Figura 3. Estructura de 3 capes d'una aplicació Web.**

### 2.1.2. Orígens, evolució i tendències.

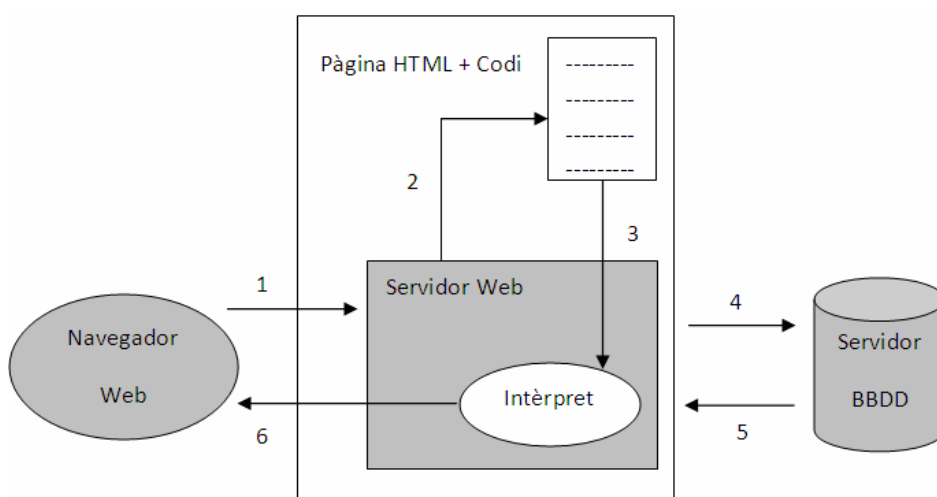
En els inicis de Internet, la Web estava formada por una sèrie de pàgines estàtiques codificades amb HTML per consultar o descarregar documents, imatges, etc. El següent pas va ser la de mostrar diferents pàgines Web amb continguts estàtics mitjançant CGI



de forma que quan des del navegador es demanava una pàgina, en el servidor Web es mirava a la seva configuració, si el que s'està demanant és una pàgina HTML normal o bé demanava l'execució d'un programa, de forma que en el cas dels CGI, el servidor l'executava passant-li les dades de la petició i generant a la sortida el nou contingut HTML en funció d'aquests paràmetres, però a la llarga provocava problemes de molta carrega i de saturació en el servidor Web, degut a la execució simultània de molts programes.

El següent pas per oferir contingut dinàmic va ser l'ús d'algun llenguatge interpretat en el costat del servidor Web, com per exemple PHP, ASP o JSP, de forma que es barreja el codi HTML i el del llenguatge del intèrpret en el servidor Web, interpretant-lo aquest en el moment en que el navegador sol·licita una pàgina dinàmica.

D'altra banda també apareixen llenguatges en el costat del client, com per exemple JavaScript, implementat com a part del navegador Web, i que permet millorar la interfície de l'usuari i les pàgines web dinàmiques. Normalment s'utilitza en les pàgines HTML per fer operacions en l'aplicació client i se interpreta en la interfície del usuari al mateix temps que les sentències es descarreguen juntament amb el codi HTML.



1. Es fa una petició d'una pàgina dinàmica.
2. El servidor Web cerca la pàgina corresponent
3. La pàgina passa per l'intèrpret per interpretar el codi que pugui haver barrejat amb l' HTML.
4. En el cas de haver de fer alguna consulta a la BBDD, la fa l'intèrpret.
5. Es retornen els resultats al servidor Web
6. A mesura que es genera la pàgina definitiva s'executa el codi, i s'envia cap al navegador per visualitzar-lo.

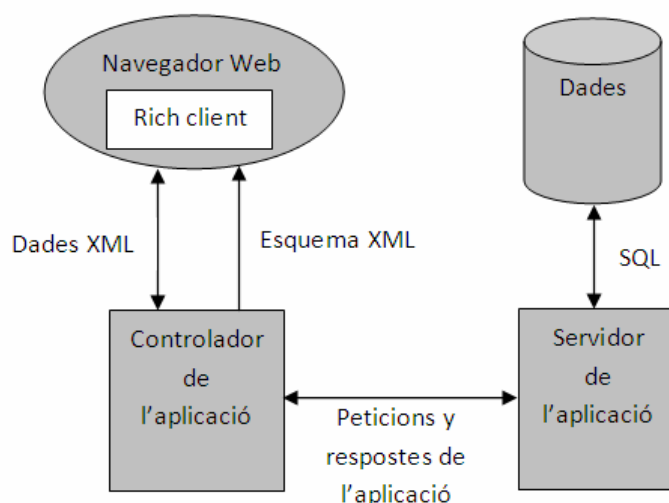
**Figura 4. Procés de generació de pàgines dinàmiques.**

De mica en mica a continuat evolucionant, tant respecte als llenguatges del costat del servidor, com en el costat del client, per exemple separant el contingut de l'estil amb CCS i millorant la presentació dels elements Web, però davant d'algunes mancances com per

exemple, la necessitat d'actualització de parts de una pàgina sense tindre la necessitat de carregar-la sencera, l'ús de capacitats multimèdia (àudio, vídeo, imatges i textos) de forma íntegra, la necessitat de recuperació de capacitats de processament en el client, o alts nivells de interactivitat, va fer aparèixer uns nous tipus d'aplicacions Web, les RIA.

Les RIA milloren l'experiència de l'usuari amb eines i funcions com copiar, tallar i pegar, redimensionar o ordenar columnes, etc, nous components, millora de la multimèdia, disminució de l'ample de banda utilitzat en l'ús de l'aplicació al permetre emmagatzemar més informació en el client, desvinculació de la presentació de la lògica, visualització i execució en múltiples plataformes i dispositius heterogenis i possibilitat d'execució online/offline.

En l'arquitectura de les RIA, la part corresponent al client conté tota la part gràfica de l'aplicació (esquema XML), limitant la comunicació entre client i servidor al traspàs de dades (normalment XML). El navegador ha de tenir instal·lat un *plugin (rich client)* que és necessari per interpretar les dades que arriben de l'aplicació. La part d'aplicació queda dividida en dues, la passarel·la (*gateway*) o controlador de l'aplicació i el servidor de l'aplicació. El controlador és l'encarregat de interactuar amb el client i transforma les dades a XML i també interactua amb el servidor per tal de sol·licitar u oferir serveis. En el servidor s'allotja l'aplicació permetent si cal, la comunicació amb altres aplicacions o *servlets*, i també s'encarrega de obtenir les dades de l'aplicació amb bases de dades o fitxers.



**Figura 5. Estructura d'una RIA**

En el següent apartat comentaré els diferents llenguatges i tecnologies existents, però cal indicar que, encara que actualment les aplicacions RIA com per exemple la tecnologia Flash de Adobe, són molt conegudes i esteses en l'àmbit de les aplicacions Web, des de finals de 2009, el W3C (*World Wide Web Consortium*), es va centrar en definir la última versió (5) d'aquest llenguatge, que com veurem posteriorment, la seva utilització aporta

moltes prestacions, entre les que cal destacar entre altres la millora en la eficiència de la reproducció de àudio i vídeo, consum de menys recursos, i gestió d'aquests amb codi obert i transparent, i sense necessitat d'instal·lar components addicionals. Sense cap dubte amb l'aparició d'aquestes especificacions es crea una alternativa [2] important de les aplicacions RIA.

## 2.2. Components, llenguatges i tecnologies de les aplicacions Web.

Tal com s'ha comentat en l'apartat anterior, en les aplicacions Web, interaccionen diferents elements com els navegadors Web dels clients, els servidors Web o els sistemes gestors de bases de dades, que fan servir diferents tecnologies i llenguatges. Es pot ampliar la documentació al respecte en

### 2.2.1. Components

Generalment en l'ús de les aplicacions Web intervenen 3 elements, el navegador Web, el servidor i el sistema gestor de bases de dades.

Actualment podem trobar uns quants navegadors Web que funcionen sobre diferents sistemes operatius com Windows, Mac OS X, o Linux. Alguns exemples son Mozilla Firefox, Chrome, Safari, Opera, o Internet Explorer.

La funció principal del navegador és la descarregar documents HTML o XHTML i mostrar-los en pantalla, així com altres tipus de documents com imatges, àudio o vídeos en diferents formats i protocols. També permeten emmagatzemar informació al disc o crear marcadors de les pàgines més visitades. En el cas de fer servir RIA, com per exemple la tecnologia Flash de Adobe, normalment cal instal·lar un *plugin* en el navegador per a la correcta visualització dels continguts (Flash Player en aquest cas).

Un altra part fonamental en les aplicacions Web, és el servidor, que s'encarrega de gestionar la comunicació entre la aplicació i els usuaris, de forma que reben les seves peticions, i retornen la informació que demanen. Normalment, els servidors Web incorporen servidors d'aplicacions que permeten interactuar amb altres sistemes com bases de dades. A dia d'avui existeixen molts servidors Web amb capacitat de generar respostes dinàmiques, alguns son lliures i altres requereixen de llicència per ser utilitzats. Alguns exemples son Apache Web Server (lliure), IIS (*Internet Information Server*) de Microsoft, Sun Java System Web Server o Zeus Web Server.

L'altre component son els sistemes gestors de bases de dades, que permeten emmagatzemar les dades per tal de poder accedir a elles posteriorment de forma ràpida i estructurada.

Aquestos sistemes son un tipus de programari molt específic, dedicat a servir de interfície entre la base de dades, l'usuari i les aplicacions que la utilitzen i estan formats pels llenguatges de definició, modificació i consulta de dades.

Alguns exemples de SGBD son els següents:

- Microsoft Access. Creat per Microsoft per a petites organitzacions, és molt conegut, la seva implementació és simple i no esta pensat per a un gran volum de dades. Es component de Microsoft Office.
- Microsoft SQL Server. Es molt més potent que Microsoft Access, permet manegar grans quantitats de dades de forma simultània i normalment s'utilitza amb grans organitzacions com bancs o administracions publiques.
- MySQL. Es un SGBD de gama mitja amb llicencia GPL que es caracteritza per la seva gran velocitat i bon rendiment sempre que el volum de dades no sigui molt gran.
- Oracle DataBase. Esta considerat un dels sistemes més complets, destacant per el suport de transaccions, estabilitat, escalabilitat i suport multiplataforma. Es el dominant en el mercat dels servidors empresarials juntament amb Microsoft SQL Server.
- eXist. És un SGBD natiu XML, de codi obert i basat en el llenguatge XQuery i proporciona un potent entorn pel desenvolupament de pàgines Web.

## 2.2.2. Llenguatges i tecnologies.

En els navegadors i els servidors Web, hem vist la necessitat de utilitzar diferents llenguatges i tecnologies que poden variar en funció de les necessitats o requeriments de les aplicacions.

### 2.2.2.1. Llenguatges del costat del client

En la part del navegador s'utilitzen diferents llenguatges:

- HTML.

Aquest llenguatge de marcat és el més utilitzat en la creació de les pàgines Web, i s'anomena així degut al seu sistema de codificació, basat en etiquetes preestablertes que afegeixen característiques de presentació i format. Aquestes etiquetes consisteixen en breus instruccions d'inici i final que els navegadors son capaços d'interpretar, i segons el codi es determina la forma en que apareixeran en la pantalla els diferents continguts, text, imatges, vídeos, etc. Aquest llenguatge des de la seva creació ha passat per diferents versions, actualment es la HTML5 [3], els diferents navegadors s'estan renovant amb noves versions per tal d'adequar-se progressivament a les noves especificacions, i per això varien les

prestacions en funció del navegador que es faci servir. Entre les novetats més destacades que presenta la última especificació, estan la possibilitat de aplicacions web sense connexió, API's d'arxius, funcions amb gràfics 2D/3D, i millores en elements bàsics, presentació, connectivitat, àudio/vídeo, emmagatzemament i rendiment.

- CSS.

És un llenguatge que descriu la presentació dels documents estructurats en fulls d'estils. El principal avantatge que presenta en el seu ús, és el de poder separar completament el contingut de l'estil (permet per exemple emmagatzemar en un sol arxiu CSS l'estil de presentació per totes les pàgines d'una aplicació Web, en un document HTML o en un fragment del mateix, o inclús en una sola etiqueta) millorant per tant el control de la presentació dels elements Web i facilitant la seva actualització.

- JavaScript.

A vegades abreujat com a JS, és un llenguatge lleuger e interpretat, orientat a objectes, conegut com el llenguatge de *script* per a pàgines web. El seu estàndard és ECMAScript, i des de 2012 tots els navegadors moderns suporten completament ECMAScript 5.1. No es necessita un kit de desenvolupament per la seva programació, ni compilar els *scripts* ni realitzar-los en documents externs al codi HTML. Un exemple senzill del seu ús el trobem en la validació dels camps dels formularis per tal de comprovar si el contingut és vàlid, abans d'enviar les dades cap al servidor web.

#### 2.2.2.2. Llenguatges del costat del servidor

En el costat del servidor per tal d'oferir contingut dinàmic s'utilitza algun llenguatge interpretat, de forma que es barreja el codi HTML i el del llenguatge del intèrpret en el servidor Web, interpretant-lo aquest en el moment en que el navegador sol·licita una pàgina dinàmica.

Alguns d'aquests llenguatges són:

- PHP.

Aquest llenguatge no necessita compilar-se per a la seva execució i la seva sintaxi és similar a llenguatges com C, Java o Perl. Com a característiques interessants, destaquen la facilitat d'implementació, ser un llenguatge multiplataforma, capacitat de connexió amb la majoria de SGBD, rapidesa, capacitat d'expansió amb nous mòduls, i la disponibilitat de molta documentació i exemples. A més és un software lliure.

- Perl.

És un llenguatge de propòsit general originalment desenvolupat per a la manipulació de text i que en l'actualitat s'utilitza per a moltes tasques com administració de sistemes, desenvolupament Web, programació en xarxa, bioinformàtica o finances. Les principals característiques són la facilitat d'ús, suporta tant la programació estructurada com la programació orientada a objectes i la programació funcional, disposa d'un poderós sistema de processament de text i molts mòduls disponibles.

- JSP.

És una tecnologia Java que permet generar contingut dinàmic. Per al seu desplegament i ús, és necessari un servidor Web compatible amb contenidors *servlets* com Apache Tomcat o Jetty. L'avantatge de JSP davant d'altres llenguatges és que el llenguatge Java és de propòsit general que sobrepassa el món Web i és apte per crear classes que manegen lògica de negoci i accés a dades. Una altra avantatge és que hereta la portabilitat de Java i és possible executar les aplicacions en diferents plataformes sense canvis.

- ASP.

Desenvolupat per Microsoft, en els seus inicis les pàgines ASP estaven formades per scripts barrejats juntament amb el codi HTML. Van ser substituïdes en el 2002 amb motiu de l'aparició de la tecnologia .NET de Microsoft. En l'actualitat ASP.NET és un *framework* per a aplicacions Web. Les seves pàgines són conegudes oficialment com formularis Web (*web forms*), i són el mitjà principal de construcció de les aplicacions. Els formularis Web estan continguts en arxius amb extensió ASPX, i típicament contenen etiquetes HTML i XHTML, així com etiquetes que defineixen controls Web que es processen en el costat del servidor i controls d'usuari on es col·loca el contingut estàtic i dinàmic requerit per la pàgina Web. ASP.NET només funciona sobre el servidor de Microsoft IIS, lo que suposa una desavantatge respecte a altres llenguatges del costat del servidor, que són executables sobre servidors més populars com Apache.

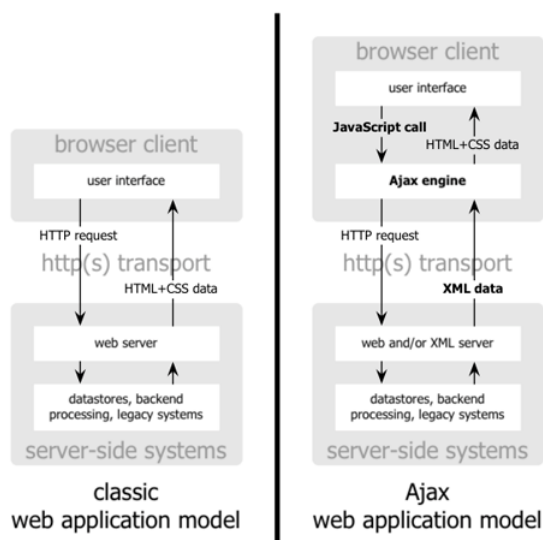
### 2.2.2.3 Tecnologies RIA.

Per a descriure les característiques que presenten les tecnologies RIA, cal comentar en primer lloc una tècnica (no és una tecnologia) anomenada AJAX.

La tècnica AJAX proposa el desenvolupament d'aplicacions Web de la següent forma:

- Presentació basada en els estàndards XHTML i CSS.
- Canvis dinàmics de la visualització del contingut mostrat i control d'events d'usuari a través del DOM.
- Intercanvi de dades i manipulació fent servir XML i XSLT.

- Recuperació de dades de forma asíncrona mitjançant XMLHttpRequest.
- Ús de JavaScript com a llenguatge d'unió de tots els components.



**Figura 6. Models de transport de les aplicacions Web, clàssic i amb AJAX.**

En el model clàssic, cada cop que volem carregar una nova pàgina, es té d'enviar una petició al servidor Web, i aquest ens retorna la nova pàgina sencera (dades més presentació), en canvi amb AJAX, al fer la petició al servidor Web ens retornarà només les dades a mostrar (en format XML), aconseguint d'aquesta manera reduir el volum de tràfic entre client i servidor sense necessitat de recarregar novament la pàgina.

Les tecnologies RIA són altament compatibles i interoperables amb la majoria de clients, sistemes operatius, servidors d'aplicacions i sistemes gestors de bases de dades, fan servir XML, i/o Java/JavaScript i altres estàndards, i busquen optimitzar i millorar la interactivitat entre el client i el servidor per reduir el tràfic de xarxa. Normalment disposen d'un entorn de desenvolupament que permet crear la capa de presentació de les aplicacions d'una manera simple.

Alguns exemples d'aquestes tecnologies són OpenLaszlo (lliure), Flex de Adobe o JavaFX de Oracle.

- OpenLaszlo [4].

OpenLaszlo és una plataforma de desenvolupament RIA de codi obert i de lliure distribució que permet desenvolupar aplicacions Web impactants i dinàmiques. Esta basat en LZX, (llenguatge orientat a objectes que fa servir XML i JavaScript), i utilitza un servidor d'aplicacions anomenat Laszlo Presentation Server (LPS) basat en Java amb la funció principal de emmagatzemar les aplicacions Web i compilar-les a fitxers SWF quan els usuaris o sol·liciten, amb la qual cosa ens caldrà tenir instal·lat el *plugin* de Adobe Flash Player en el navegador Web. El

LPS s'executa en un servidor d'aplicacions J2EE estàndard o en un contenidor de *servlets* Java i pot ser executat en qualsevol sistema operatiu.

El servidor OpenLaszlo consta de 3 subsistemes principals, el compilador de interfícies, (format per el compilador LZX, el transcodificador de medis i la cache), el gestor de connexions permanents i el gestor de dades. El compilador d'interfícies converteixen els arxius d'origen en arxius executables (SWF) i els ofereixen en un arxiu binari que s'executa en el navegador del client (com per exemple Flash o J2ME), o com un JavaScript (DHTML) executat pel mateix navegador. El transcodificador de medis converteix tots els tipus de recursos de medis en un únic format, presentant els diferents tipus de media (JPEG, GIF, PNG, MP3, TrueType y SWF animació) de manera única en la mateixa pantalla. El gestor de dades permet a les aplicacions Laszlo interactuar amb bases de dades, serveis Web XML i fitxers executables en un servidor Web. Finalment el gestor de connexions persistents gestiona l'autenticació i la missatgeria en temps real per les aplicacions que o necessiten.

L'arquitectura en el costat del client es basa en les classes de la Laszlo Foundation, de manera que cada vegada que un client invoca una aplicació, es descarreguen també les biblioteques necessàries i la font en temps d'execució, de manera que el client manté en tot moment la connexió amb el servidor. Consta de 3 subsistemes, el sistema d'events, el carregador, enllaçador de dades i el sistema de disposició i animació.

Hi ha un entorn de desenvolupament anomenat IDE4Laszlo creat per la empresa IBM que es de lliure distribució, i que consisteix en un *plugin* que se instal·la en el editor de programació Eclipse. És interessant una funcionalitat anomenada SOLO que permet precompilar la aplicació LZX en un fitxer SWF per inserir-lo en una pàgina HTML i executar-lo en qualsevol servidor Web sense la necessitat de tenir instal·lat el servidor de presentació Laszlo.

- Adobe Flex [5]

Flex, era un producte de Macromedia (fins a 2005) que agrupava una sèrie de tecnologies per donar suport al desplegament i desenvolupament de RIA, basades en la plataforma propietària Flash. Inicialment va ser alliberat com una aplicació de la J2EE o biblioteca de JSP que compilava un propi llenguatge de etiquetes XML (MXML) per definir la interfície d'usuari i un llenguatge *script* orientat a objectes anomenat ActionScript que permet executar aplicacions Flash (SWF). Posteriorment les darreres versions de Flex suporten la creació d'arxius estàtics que son compilats i poden ser distribuïts sense necessitat d'una llicència de servidor.



Flex disposa de gran varietat de components i característiques que aporten funcionalitats com serveis Web, objectes remots, columnes ordenables, gràfiques, efectes d'animació i altres interaccions. El servidor de presentació Flex s'executa en el servidor de l'aplicació i té capacitat de integració i gestió per a les aplicacions Flex. Aquesta capacitat d'integració facilita l'aprofitament de codi fent servir serveis web, accés a objectes de Java o XML.

El model basat en components facilita la creació de les aplicacions Flex, de manera que els desenvolupadors poden fer servir els components inclosos o estendre'ls per afegir noves propietats i mètodes, així com crear nous components.

Hi ha un entorn de desenvolupament anomenat Flash Builder que juntament amb Flex, formen una solució completa, però no és lliure, sinó que és propietari de Adobe.

- JavaFX

JavaFX és una extensió a la plataforma Java que ofereix interactivitat, animació i programació compatible amb AJAX, Flash de Adobe i la nova plataforma Silverlight de Microsoft. Permet als desenvolupadors crear i desplegar aplicacions RIA que se comporten de forma consistent en diferents plataformes. Esta construït sobre la tecnologia Java i proporciona un gran conjunt de API's de gràfics i continguts multimèdia d'alt rendiment amb acceleració per hardware de gràfics.

JavaFX en lloc d'utilitzar al navegador com a contenidor del codi JavaScript, les aplicacions fan servir els dispositius de seguretat de Java SE per a controlar el accés a disc de les aplicacions. Al córrer sobre el client i no dependre del codi que travessa la xarxa, es poden utilitzar aplicacions AJAX, com les de Google Apps per exemple, en mode *offline*.

Un exemple de entorn de desenvolupament per a JavaFX que es pot utilitzar via *webstart* és JFXBuilder y també es poden utilitzar els entorns de desenvolupament per a Java como NetBeans IDE.

## 2.3. Els continguts multimèdia i la transmissió de vídeo.

Un cop definides les diferents parts que formen part de les aplicacions Web, com interactuen, quins son els llenguatges que podem fer servir tant en el costat del client com del servidor, així com les possibilitats que ofereixen les tecnologies RIA, donat que en la aplicació a desenvolupar ha de permetre la reproducció de vídeos, cal veure quins son els diferents formats existents, com es genera el corrent i la recepció del vídeo, així com alguns dels serveis de vídeo sota demanda existents.

## 2.3.1. Formats i contenidors multimèdia.

Seguidament es comenten els diferents formats de àudio i vídeo que existeixen actualment, així com de contenidors multimèdia.

### 2.3.1.1. Formats de àudio.

- MPEG (*Moving Picture Expert Group*)

És el resultat del treball de membres de diferents indústries i universitats que s'encarrega de desenvolupar estàndards de codificació d'àudio i vídeo i de la creació d'altres normes auxiliars com les metadades. Hi ha diferents possibilitats respecte als codecs de àudio de MPEG:

- MP3 o H.260: MP3 és l'acrònim de MPEG-1 *layer 3*. És un format d'àudio digital comprimit amb pèrdua. Va ser el primer format d'àudio popularitzat gràcies a Internet, ja que va fer possible intercanviar fitxers musicals.

- MPEG-2 Àudio (part 3): Introdueix algunes millores respecte MPEG-1, com per exemple codificar fins a 6 canals i permet que descodificadors MPEG-1 puguin interpretar el seu format.

- MPEG-4 Àudio (part 3): Consisteix en diferents components, la majoria dels quals estan programats en un llenguatge anomenat SAOL. Basat en la compressió de senyals d'àudio de codificació perceptual incloent variacions del *Advanced Audio Coding (AAC)*.

- Vorbis

Codificador d'àudio lliure de compressió amb pèrdues. Forma part del projecte Ogg i llavors és anomenat Ogg Vorbis. Permet escalar sobre una gran varietat de *bitrates*.

- WMA (Windows Media Audio)

Format d'àudio dissenyat per Microsoft. És inferior tècnicament a formats com MP3 o Vorbis.

- AAC

Format d'àudio digital comprimit amb pèrdua, va ser dissenyat per substituir al MP3. És un format utilitzat per aplicacions com Nero, iTunes o Winamp.

- Real Audio

Format propietari de Real Networks. Es pot trobar amb diferents rangs de taxa de bits per poder treballar en diferents entorns i està preparat per ser utilitzat com un format d'àudio *streaming*. Moltes ràdios per Internet el fan servir.

### 2.3.1.2. Formats de vídeo.

- MPEG

Hi ha diferents possibilitats:

- MPEG-1 (part 2): Codec de vídeo per a senyals no entrellaçades (progressives).

- MPEG-2 (part 2) o H.263: Codec de vídeo per difusió de qualitat televisiva. Utilitzat per serveis de televisió per satelit, senyals de televisió digital per cable i per als DVD.

- MPEG-4 (part 10) AVC o H.264: Norma que defineix un vídeo d'alta compressió. La intenció del projecte va ser la de crear un estàndard capaç de proporcionar una bona qualitat d'imatge amb mida de memòria notablement inferior als estàndards previs (MPEG-2 part 2), amés de no incrementar la complexitat del seu disseny.

- Format Flash Video.

Son arxius FLV i contenen flux de vídeo que son una variable del estàndard H.263 anomenats *Sorenson Park*.

- Theora

Codec de vídeo lliure desenvolupat per la fundació Xiph.org com a part del projecte Ogg. Basat en la codificació de vídeo amb pèrdues que generalment es troba dins el contenidor Ogg, combinat amb el format d'àudio Vorbis.

- VP8

Codec de codi obert creat per On2Technologies (posteriorment comprada per Google). Moltes empreses y navegadors aposten per que aquest sigui el format multimèdia estàndard en el llenguatge Web HTML5.

- WMV (Windows Media Video)

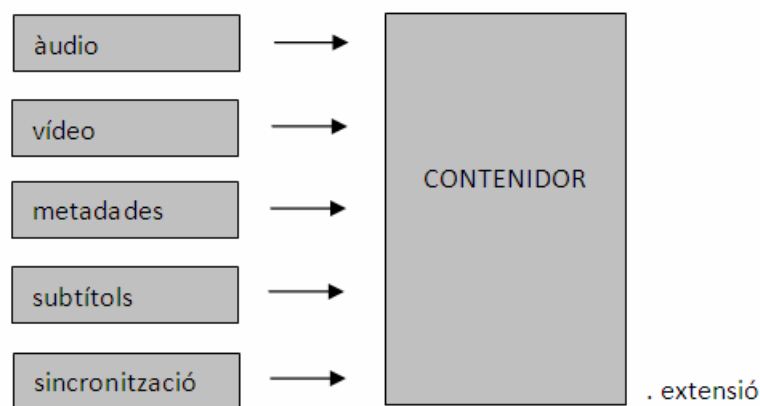
Nom genèric del conjunt d'algorismes de compressió del grup de tecnologies de vídeo desenvolupat per Microsoft i que es combina generalment amb WMA.

- Real Video

Format de vídeo propietari desenvolupat per Real Networks. Normalment va lligat amb l'àudio de Real Audio dins del contenidor Real Media (.rm) i suporta streaming de vídeo per als diferents usos a Internet.

### 2.3.1.3. Els contenidors multimèdia.

Els contenidors multimèdia son un tipus d'arxiu que emmagatzema informació de vídeo, àudio, subtítols, metadades e informació de sincronització seguint un format establert en la seva especificació.



**Figura 7. Contenedors multimèdia.**

Alguns exemples d'arxius contenidors que podem trobar actualment són:

- MPEG
  - MPEG-1 (part 1): Conegut com a PS (*Program Stream*).
  - MPEG-2 (part 1): Conegut pel nom de TS (*Transport Stream*). En aquesta versió, el PS està més destinat a contenir per corrents multimèdia dels DVD's.
  - MPEG-4 (part 14): Normalment s'anomena mp4. Va ser dissenyat per transportar àudio i vídeo (especialment si estan definits per MPEG), encara que també suporta altres corrents de dades com subtítols o imatges estàtiques. Permet streaming per Internet.
- Contenedor del Flash Video (SWF).
 

Format d'arxiu propietari de gràfics vectorials creat per Macromedia (actualment Adobe). Es utilitza per aplicacions RIA i per transmetre vídeo per Internet fent servir Adobe Flash Player en el navegador Web, amb FLV com format de vídeo i MP3 com format d'àudio.
- OGG
 

Format desenvolupat per la fundació Xiph.org i a la vegada el seu format natiu. El format està lliure de patents i obert, al igual que tota la tecnologia de Xiph.org. Dissenyat per donar un alt grau d'eficiència en l'streaming i la compressió d'arxius.
- WebM
 

És un contenidor obert i lliure desenvolupat per Google orientat a fer-lo servir amb HTML5. És un projecte de software lliure sota una llicència permissiva similar a la llicència BSD, i està format pel codec de vídeo VP8 i el codec de àudio Vorbis.
- AVI (*Audio Video Interleave*)

Definit per Microsoft per la seva tecnologia Video for Windows en 1992, va ser millorat posteriorment mitjançant les extensions de forma del grup OpenDML de la companyia Matrox. Aquestes extensions es coneixen com a AVI 2.0.

### 2.3.2. Transmissió de vídeo.

En la transmissió de vídeo sota demanda, no fa falta la descarrega completa de l'arxiu per poder veure'l, de manera que els usuaris s'estalvien la espera de que l'arxiu sigui transmès totalment.

#### 2.3.2.1. Generació del corrent de vídeo.

Per generar un corrent de vídeo sota demanda es necessita treballar amb un servidor streaming. El servidor s'encarrega de negociar amb el reproductor de l'usuari la velocitat més adequada per l'entrega de dades. Servidor i client mantenen una comunicació constant durant aquest procés, de manera que el servidor pot respondre a qualsevol tipus de problemes que pateixi la xarxa o de petició per part del client. Al moment de transferir les dades hi ha 3 modalitats, *unicast* (enviament de un emissor a un receptor), *multicast* (a múltiples destinataris simultàniament) o *broadcast* (a tots els nodes de la xarxa). Algun del programari que podem trobar són per exemple, *Windows Media Services*, *Quick Time Streaming Server*, *Flash Media Server* o *VLC*.

#### 2.3.2.2. Reproductors del corrent de vídeo.

Els reproductors del corrent de vídeo són els encarregats d'interpretar les dades en streaming per reproduir els arxius de forma adequada, i normalment utilitzen un *plugin* dels navegadors Web o directament com una aplicació. El procés s'inicia quan el reproductor comença a rebre el fitxer i el construeix al *buffer* emmagatzemant la informació. Quan s'omple amb una petita part de l'arxiu, el reproductor comença a reproduir-lo i a la vegada el continua descarregant. El sistema està sincronitzat de manera que quan acaba la transferència, acaba la reproducció. Si en el procés la connexió pateix un descens de velocitat s'utilitza la informació del buffer de manera que es pot suportar temporalment aquest descens. Alguns dels reproductors que podem trobar són *Real Player*, *Windows Media Player*, *Quick Time Player*, *Adobe Flash Player* o *VLC*.

#### 2.3.2.3. Serveis de vídeo sota demanda.

Actualment, els serveis de vídeo sota demanda tenen molta popularitat, no només pel fet de l'ús de la reproducció de vídeos per Internet, sinó que s'utilitzen en diferents àrees com en l'art i disseny multimèdia (Promoció musical o informatius), comunicacions corporatives, publicitat, docència o compartició de vídeos.

Les aplicacions en aquest últim camp permet que qualsevol usuari pugui pujar qualsevol vídeo per a la seva reproducció a la xarxa. Alguns exemples són *YouTube*, *Google Video*

o *Vimeo*. També hi ha actualment moltes cadenes de televisió que ofereixen en les seves pàgines Web la possibilitat de poder veure series, programes o esdeveniments esportius en diferit, algunes gratuïtes i altres de pagament.

#### 2.3.2.4. Reproducció de vídeo amb HTML5

Amb les noves especificacions del W3C pel que fa a HTML5, s'intenta estandarditzar el llenguatge de marques amb noves prestacions i possibilitats per tal de fer front a aquest tipus d'aplicacions. Respecte als continguts de vídeo en lloc de implementar la representació típica amb el format de Adobe Flash, intentaré fer servir les noves possibilitats que ofereix HTML5 respecte a l'ús i representació de vídeo. Aquest aspecte ha evolucionat molt des de finals de 2009 fins ara, i cal tenir-ho en compte donat que actualment depenent del navegador o dels contenidors multimèdia, podem visualitzar o no els continguts.

Les següent figura mostra informació [6] sobre el augment en el compliment de les especificacions per part dels diferents navegadors en les seves darreres versions (sobre un màxim de 555 punts), i la compatibilitat dels navegadors amb les noves característiques d'àudio i vídeo de HTML5.



**Figura 8. Evolució del compliment de HTML5 segons els navegadors.**

En la següent figura podem veure que la majoria de navegadors, a excepció de les versions 7 i 8 de Internet Explorer suporten elements com vídeo, àudio, detecció de codecs, gràfics en 2D o la presentació inicial d'una imatge abans de mostrar el vídeo.

**FEATURES**  
Select up to five features and immediately see how well it is supported by each browser

	video element	Poster image support	Codec detection	Canvas 2D graphics	audio element
<b>desktop browsers</b>					
Chrome 30 »	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
Chrome 32 »	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
Firefox 24 »	Yes ✓	Yes ✓	Buggy ⚠	Yes ✓	Yes ✓
Firefox 26 »	Yes ✓	Yes ✓	Buggy ⚠	Yes ✓	Yes ✓
Internet Explorer 7 »	No ✗	No ✗	No ✗	No ✗	No ✗
Internet Explorer 8 »	No ✗	No ✗	No ✗	No ✗	No ✗
Internet Explorer 9 »	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
Internet Explorer 10 »	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
Internet Explorer 11 »	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
Opera 12.00 »	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
Opera 12.10 »	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
Opera 16 »	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
Opera 17 »	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
Safari 6.0 »	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓
Safari 7.0 »	Yes ✓	Yes ✓	Yes ✓	Yes ✓	Yes ✓

**Figura 9. Compatibilitat navegadors i àudio/vídeo de HTML5.**

## 2.4. Llenguatges i tecnologies escollides.

En la part final de l'estat de l'art, caldrà triar les opcions més adients al projecte. D'una banda per a l'aplicació Web a desenvolupar, no és necessària molta interacció amb l'usuari, ni cal una interfície d'usuari especialment "rica", donat que es tracta d'una aplicació per a una Intranet, en la que els possibles usuaris (s'estimen com a molt uns 50 autoritzats) accediran per consultar documents, i que a més difícilment coincidiran en el temps. Això deixa una mica de banda el possible ús de les tecnologies RIA, o la necessitat d'un servidor Web d'altres prestacions. Un altre factor molt important és l'ús d'eines i tecnologies de codi lliure, per la qual cosa serà important triar bona combinació entre llenguatge del costat del servidor, servidor Web i SGBD. Un bon exemple és l'ús de PHP+Apache+MySQL, on podem trobar plataformes que les integren de forma conjunta, i que ofereixen a més diferents eines i opcions d'administració i escalabilitat totes elles gratuïtes.

### 2.4.1. Servidor Web Apache

El servidor web Apache [7] és actualment un dels més utilitzats al món. Esta basat en el NCSA Server i va sorgir com a fruit d'ampliacions i versions d'aquest, fins a que es va convertir en un producte diferent. Molts servidors d'aplicacions l'inclouen com una capa més dintre dels seus productes com Oracle o IBM.

En la actualitat el seu desenvolupament depèn de la *Apache Software Foundation*. És un projecte open-source, és a dir, el seu codi font esta totalment disponible i tothom pot modificar-lo i fer la seva aportació, evidentment sota la supervisió de la *Apache Software Foundation*, sent una de les raons que l'han fet tant popular és la seva gratuïtat.

Esta dissenyat de forma modular, i es poden trobar una gran quantitat de mòduls que li donen una gran flexibilitat. Alguns d'aquestos mòduls venen activats amb la distribució bàsica del servidor, i altres es poden activar de ser necessaris. Alguns d'aquestos s'agrupen en termes com autenticació, control d'usuaris, registre d'accessos o contingut dinàmic, i concretament dintre d'aquest últim tenim el modul `mod_cgi` per executar CGI, el modul `mod_PHP` per fer servir el llenguatge PHP, o el `mod_jk` per comunicar-se amb el servidor de Servlets/JSP Tomcat. Es independent de la plataforma, és a dir, funciona sobre diferents plataformes com Unix, Linux, Windows, OS2, etc.

### 2.4.2. SGBD MySQL

MySQL [8] és un SGBD relacional molt ràpid i robust, llicenciat sota la GPL (*GNU Public License*). Va ser creat per la empresa sueca MySQL AB, que manté el copyright del codi font del servidor SQL, així com de la marca.

Aquest gestor de base de dades és probablement el gestor més utilitzat en el món del software lliure gracies a la seva rapidesa i facilitat d'ús. Aquesta gran acceptació es deguda en part, a l'existència d'infinat de llibreries i altres eines que permeten el seu ús mitjançant diferents llenguatges de programació, a més de la facilitat d'instal·lació i configuració. Aprofita la potencia dels sistemes multiprocés gracies a la seva implementació *MultiThread* i destaca també per ser un sistema flexible que suporta diferents plataformes i sistemes operatius.

### 2.4.3. El llenguatge del costat del servidor, PHP.

En la elecció del llenguatge de script del costat del servidor es va descartar l'ús de ASP.NET per ser una tecnologia propietària de Microsoft, que a més funciona sobre el servidor web IIS també d'aquesta empresa, i una de les principals premisses era la elecció de llenguatges de codi obert. Respecte a l'opció de JSP, és una tecnologia Java que permet generar contingut dinàmic en la que per al seu desplegament i ús, és necessari un servidor Web compatible amb contenidors *Servlets* com Apache Tomcat o Jetty. L'ús de JSP davant d'altres llenguatges presenta com avantatge que el llenguatge Java és de propòsit general que sobrepassa el món Web i és apte per crear classes que manegen lògica de negoci i accés a dades, però en el cas de l'aplicació que es vol implementar, les següents característiques han fet seleccionar a PHP [9] com a llenguatge del costat del servidor escollit.

PHP no necessita compilar-se per a la seva execució i la seva sintaxi és similar a llenguatges com C, Java o Perl. És un software lliure i gratuït eficient degut als pocs recursos que consumeix el servidor, de forma que amb un equip senzill es poden desenvolupar molt bones aplicacions. Destaca la facilitat d'implementació, la seva portabilitat i multiplataforma, capacitat de connexió amb la majoria de SGBD com per exemple el escollit MySQL, rapidesa, gran capacitat d'expansió del seu potencial amb



nous mòduls, i la disponibilitat d'una documentació oficial molt extensa a la seva pàgina Web, on s'inclouen exemples de totes les seves funcions.

#### 2.4.4. Llenguatges i llibreries del costat del client.

En el costat del client, és obligatori l'ús del llenguatge de marcat HTML, actualment en la seva versió HTML5. Aquest llenguatge basat en etiquetes preestablertes que afegeixen característiques de presentació i format, consisteix en breus instruccions d'inici i final que els navegadors son capaços d'interpretar, i segons el codi es determina la forma en que apareixeran en la pantalla els diferents continguts, text, imatges, vídeos, etc. En la seva última versió, els diferents navegadors s'estan renovant amb noves versions per tal d'adequar-se progressivament a les noves especificacions, i per això varien les prestacions en funció del navegador que es faci servir. Entre les novetats més destacades que presenta la última especificació, estan la possibilitat de aplicacions web sense connexió, API's d'arxius, funcions amb gràfics 2D/3D, i millores en elements basics, presentació, connectivitat, àudio/vídeo, emmagatzemament i rendiment.

Un altre llenguatge que s'utilitzarà en el desenvolupament de l'aplicació Web, és el CSS, actualment en la seva versió CSS3. Aquest llenguatge descriu la presentació dels documents estructurats en fulls d'estils. El principal avantatge que presenta en el seu ús, es el de poder separar completament el contingut de l'estil (permet per exemple emmagatzemar en un sol arxiu CSS l'estil de presentació per totes les pàgines d'una aplicació Web, en un document HTML o en un fragment del mateix, o inclús en una sola etiqueta) millorant per tant el control de la presentació dels elements Web i facilitant la seva actualització posterior.

En el costat del client també s'utilitzarà JavaScript, llenguatge lleuger e interpretat orientat a objectes, conegut com el llenguatge de *script* per a pàgines web. El seu ús ens pot ser útil per exemple per tal de comprovar si el contingut dels formularis es vàlid, abans d'enviar les dades cap al servidor web. D'altra banda, existeixen diferents biblioteques per a JavaScript que permet desenvolupar aplicacions Web, de manera que no cal preocupar-se per les diferències i les incompatibilitats entre navegadors. Aquestes llibreries permeten interactuar amb el DOM i per exemple seleccionar un conjunt de nodes i aplicar-li estils CSS, generar nou contingut, interactuar amb l'usuari mitjançant el sistema d'esdeveniments, capturant les accions de l'usuari i processant-les. De entre les llibreries existents treballarem amb *jQuery* que és una de les més esteses, estimant que 3 de cada 4 llocs web que utilitzen llibreries JavaScript la fan servir. També la utilitzen empreses com Amazon, Microsoft o Twitter. També s'utilitzaran diferents components visuals com calendaris, o components interactius i efectes amb l'ús de les *jQueryUI*, que formen part d'un projecte paral·lel a *jQuery*.

## 3. Disseny

Per començar el desenvolupament de l'aplicació, s'ha seguit a grans trets un mètode simplificat [10] que cobreix les cinc fases principals (definició de requisits, disseny, implementació, proves i lliurament) i una altra de permanent que es segueix en cadascuna de les anteriors a mesura que es va desenvolupant. En la fase de disseny s'apliquen diferents tècniques en les diferents fases [11]. Concretament, la fase de disseny comença detallant les funcionalitats desitjades de l'aplicació per continuar amb l'especificació dels actors i usos. Posteriorment es continua amb el disseny de la base de dades (conceptual i lògic, deixant el físic per a la fase d'implementació), es mostrarà l'esquema d'interfície de l'usuari i per últim veurem els llenguatges, tecnologies i aplicacions escollides.

### 3.1. Requeriments i funcionalitats de l'aplicació Web.

A continuació es descriuen els requeriments i funcionalitats principals de l'aplicació:

- La aplicació s'instal·larà en un ordinador de una Intranet de la empresa i serà accessible per diferents tipus d'usuaris amb diferents permisos per tal de poder permetre la seva funcionalitat principal, veure continguts com vídeos, imatges i documents relacionats amb el món ferroviari.
- Els continguts faran referència a localitzacions concretes (estacions o trajectes) on estan situats. Aquestes ubicacions s'han de poder gestionar (altes, baixes i modificació de dades).
- Els continguts faran referència a una instal·lació o element del àmbit ferroviari englobat dintre de una especialitat concreta de la infraestructura ferroviària (senyalització, telecomunicacions, infraestructura, via o electrificació) o a un conjunt d'elles. Aquestos elements s'han de poder gestionar (altes, baixes i modificacions).
- L'aplicació ha de permetre fer cerques com a mínim en funció o bé del tipus de contingut, o en funció de la seva ubicació. En aquest últim cas, el sistema ha de permetre per exemple, mostrar un llistat amb els continguts (de qualsevol tipus) que hi ha en una estació o trajecte entre estacions concrets per posteriorment poder-los visualitzar i/o gestionar.
- L'aplicació ha de permetre donar de alta qualsevol dels tres tipus de continguts, fent referència obligatòriament a una localització i a una instal·lació. El sistema ha de permetre donar d'alta un contingut concret (vídeo, foto o document), a un conjunt de fotos o a varis documents, sempre que facin referència a una mateixa localització i instal·lació, quedant exclosa inicialment la possibilitat de donar d'alta més d'un vídeo alhora per motius de volum de dades.

- A l'aplicació Web només tindran accés les persones autoritzades i consistirà en el fet de tenir un nom d'usuari i una contrasenya vàlids en el sistema.
- L'aplicació ha de permetre a qualsevol usuari validat en l'aplicació modificar la seva pròpia contrasenya.
- Hi hauran 3 rols diferents, amb el nivell més basic només podran visualitzar els diferents continguts, els usuaris amb un nivell intermedi a més de gestionar els propis continguts (altes, baixes, o modificacions de les dades associades), podran gestionar els elements amb els que poden estar relacionats, així com les ubicacions. Finalment els de majors privilegis han de poder a més, administrar els usuaris de l'aplicació (alta, modificació de dades o rols i baixes).
- Els vídeos es mostraran en el navegador sense fer servir cap *plugin* extern.

## 3.2. Casos d'ús de l'usuari.

Per expressar gràficament les relacions entre els diferents usos que tindrem i els seus participants o actors farem servir un diagrama UML, concretament el de casos d'ús on veurem les opcions que ofereix l'aplicació a cada tipus d'usuari i s'explicarà posteriorment amb més detall cadascun dels casos d'ús amb la ajuda de taules CRC.

### 3.2.1. Diagrama.

En la següent figura es mostra el diagrama amb les opcions dels usuaris de l'aplicació:

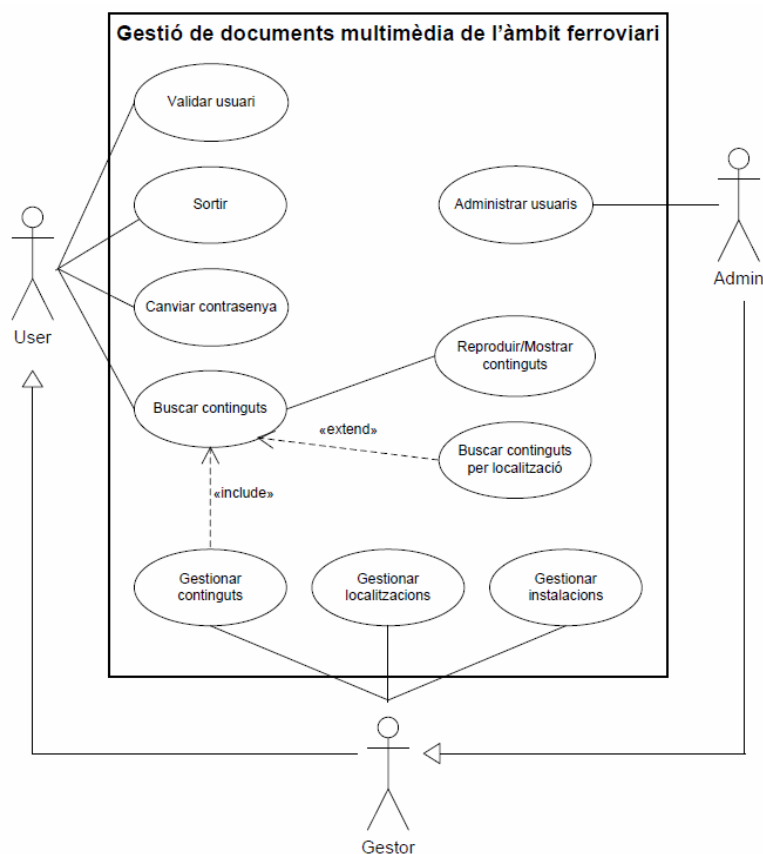


Figura 10. Diagrama de casos d'ús de l'aplicació.

Per major claredat, hi ha diferents casos d'ús que es descriuen de forma general, i que a la vegada es poden descompondre en altres més específics.

Concretament:

- El cas d'ús “Administrar usuaris” en alt nivell, es desglossaria en “Alta d'usuari”, “Eliminar usuari” i “Modificar usuari”.
- “Gestionar instal·lacions”, es descompon en “Alta d'instal·lació”, “Eliminar instal·lació” i “Modificar instal·lació”.
- “Gestionar localitzacions”, es desglossa en “Alta de localització”, “Eliminar localització” i “Modificar localització”.
- “Gestionar continguts” és descompon en “Alta de contingut”, “Eliminar contingut” i “Modificació de contingut”.

El cas d'ús “Reproduir/Mostrar continguts” es podria descompondre pel tipus de contingut, degut a la particularitat de que es podran mostrar més d'una imatge en la pàgina web, però en principi només es detallarà en la descripció del cas d'ús. Igualment en el cas “Alta de contingut”, es podria descompondre pel tipus de contingut, degut a la particularitat de que es podrien donar d'alta d'un sol cop un conjunt d'imatges o documents que facin referència a una instal·lació i localització concreta, però en principi no es descompon i es detallarà igualment en el punt 3.2.2.2.

### 3.2.2. Casos d'ús de l'aplicació

En aquest apartat es detallen els actors que interactuaran amb l'aplicació i s'especifiquen els diferents casos d'ús de l'aplicació.

#### 3.2.2.1. Actors

Tal com veiem en la figura 1, es definiran tres tipus o rols d'usuaris diferents:

- User: Correspon amb el rol d'usuari amb menys privilegis tenint en compte els requeriments definits en l'apartat 2.1.
- Gestor: A més de tindre els privilegis anteriors, pot gestionar continguts, elements i localitzacions. Quan parlem de gestió, estem incloent les accions bàsiques de alta, modificació i baixa, a més de les de consulta i cerca del nivell anterior.
- Admin: Correspon amb l'usuari administrador del web. És el que té més privilegis i a més de les funcionalitats del gestor i del user, també pot administrar els usuaris (alta, baixa i modificació de les dades dels usuaris, incloent les contrasenyes).

### 3.2.2.2. Detall de cada cas d'ús.

Per a cada cas d'ús es mostra, el nom, la descripció, el actor o actors que intervenen, les pre condicions, el flux o curs normal d'esdeveniments i els alternatius en el cas d'haver-los, les entrades i les sortides.

**Taula 2. Casos d'ús de l'aplicació.**

Nom	<b>Validar usuari</b>
Descripció	Validació d'un usuari per accedir a l'aplicació.
Actors	User, Gestor i Admin.
Flux normal	<ol style="list-style-type: none"> <li>1. L'usuari introdueix el nom d'usuari i la contrasenya.</li> <li>2. Clica el boto "Login"</li> <li>3. El sistema compara les dades introduïdes amb les del sistema.</li> <li>4. Es valida l'usuari.</li> </ol>
Flux alternatiu	<ol style="list-style-type: none"> <li>3a. El nom d'usuari no existeix</li> <li>3b. La contrasenya no coincideix amb la del usuari.</li> <li>4a,b El sistema mostra un missatge a l'usuari y continua en el punt 1 del flux normal</li> </ol>
Entrada	Credencials de l'usuari
Sortida	Es valida l'usuari en el sistema començant la sessió de l'usuari i el sistema mostra la pantalla de inici.

Nom	<b>Sortir</b>
Descripció	Desconnexió de l'usuari de l'aplicació.
Actors	User, Gestor i Admin.
Pre condicions	L'usuari esta validat en l'aplicació.
Flux normal	1. L'usuari selecciona l'enllaç de desconnectar
Flux alternatiu	1a. L'usuari tanca el navegador web
Sortida	S'elimina la sessió de l'usuari i es mostra la pantalla de login

Nom	<b>Canviar contrasenya</b>
Descripció	Modificar la contrasenya del propi usuari.
Actors	User, Gestor i Admin.
Pre condicions	L'usuari esta validat en l'aplicació.
Flux normal	<ol style="list-style-type: none"> <li>1. L'usuari selecciona l'enllaç de canviar contrasenya.</li> <li>2. El sistema mostra el formulari per introduir la contrasenya</li> <li>3. L'usuari introdueix la contrasenya actual i la nova dues vegades.</li> <li>4. El sistema modifica la clau de l'usuari en el sistema</li> <li>5. El sistema mostra un missatge indicant que s'ha canviat correctament la contrasenya.</li> <li>6. El usuari confirma missatge.</li> </ol>
Flux alternatiu	<ol style="list-style-type: none"> <li>3a. La contrasenya actual o la proposada no es correcta</li> <li>4a. El sistema mostra un missatge a l'usuari y continua en el punt 2 del flux normal</li> </ol>
Entrada	Contrasenya actual i la nova (dues vegades)
Sortida	Es modifica la contrasenya i continua en la pantalla d'inici.

Nom	<b>Alta d'usuari</b>
Descripció	El administrador dona d'alta un nou usuari.
Actors	Admin.
Pre condicions	L'administrador esta validat en l'aplicació.
Flux normal	<ol style="list-style-type: none"> <li>1. Selecciona l'opció d'usuaris del menú.</li> <li>2. El sistema mostra la pantalla de gestió d'usuaris amb el formulari per introduir les dades</li> <li>3. Introdueix les dades del nou usuari i clica el botó de "Alta".</li> <li>4. El nou usuari es dona d'alta en el sistema.</li> <li>5. Es mostra missatge confirmant l'alta.</li> </ol>
Flux alternatiu	<ol style="list-style-type: none"> <li>4a. Ja existeix aquest usuari i no es pot donar d'alta.</li> <li>5a. Es mostra missatge per informar i continua en 2. del flux normal</li> </ol>
Entrada	Dades del nou usuari, la contrasenya i el perfil assignat
Sortida	Es dona d'alta l'usuari i continua en la pantalla de gestió d'usuaris.

Nom	<b>Modificar usuari</b>
Descripció	El administrador modifica les dades d'un usuari.
Actors	Admin.
Pre condicions	L'administrador esta validat en l'aplicació.
Flux normal	<ol style="list-style-type: none"> <li>1. Selecciona l'opció d'usuaris del menú.</li> <li>2. Es mostrarà per pantalla la gestió d'usuaris amb el llistat d'usuaris.</li> <li>3. Es selecciona l'usuari i es modifiquen les dades.</li> <li>4. Clica el boto de "Modificar"</li> <li>5. Es modifiquen les dades de l'usuari en el sistema.</li> <li>6. Es mostra missatge confirmant la modificació de dades.</li> </ol>
Flux alternatiu	<ol style="list-style-type: none"> <li>5a. Ja existeix aquest nom d'usuari i no es pot assignar.</li> <li>6a. Es mostra missatge per informar i continua en 2. del flux normal</li> </ol>
Entrada	Dades modificades del usuari escollit.
Sortida	Es modifiquen les dades i continua en la pantalla de gestió d'usuaris.

Nom	<b>Eliminar usuari</b>
Descripció	El administrador dona de baixa un usuari de l'aplicació.
Actors	Admin.
Pre condicions	L'administrador esta validat en l'aplicació.
Flux normal	<ol style="list-style-type: none"> <li>1. Selecciona l'opció d'usuaris del menú.</li> <li>2. Es mostrarà per pantalla la gestió d'usuaris amb el llistat d'usuaris.</li> <li>3. Es selecciona l'usuari i clica el boto de "Eliminar"</li> <li>4. El sistema elimina l'usuari de l'aplicació.</li> <li>5. Es mostra missatge confirmant la baixa.</li> </ol>
Entrada	Usuari seleccionat.
Sortida	S'elimina l'usuari i continua en la pantalla de gestió d'usuaris.

Nom	<b>Alta de instal·lació</b>
Descripció	Donar d'alta una nova instal·lació o element.
Actors	Gestor, Admin.
Pre condicions	L'usuari esta validat en l'aplicació.
Flux normal	<ol style="list-style-type: none"> <li>1. Selecciona l'opció d'instal·lacions del menú.</li> <li>2. El sistema mostra la pantalla de gestió d'instal·lacions.</li> <li>3. L'usuari introdueix les dades del nou element clica el botó de "Alta".</li> <li>4. La nova instal·lació es dona d'alta en el sistema.</li> <li>5. Es mostra missatge confirmant l'alta.</li> </ol>
Flux alternatiu	<ol style="list-style-type: none"> <li>4a. Ja existeix aquest element i no es pot donar d'alta.</li> <li>5a. Es mostra missatge per informar i continua en 2. del flux normal</li> </ol>
Entrada	Dades de la nova instal·lació, especialitat i element
Sortida	Es dona d'alta la instal·lació i continua en la pantalla de gestió d'instal·lacions.

Nom	<b>Modificar instal·lació</b>
Descripció	Modificar les dades d'una instal·lació concreta.
Actors	Gestor, Admin.
Pre condicions	L'usuari esta validat en l'aplicació.
Flux normal	<ol style="list-style-type: none"> <li>1. Selecciona l'opció d'instal·lacions del menú.</li> <li>2. Es mostra la pantalla de gestió d'instal·lacions amb el llistat d'instal·lacions.</li> <li>3. L'usuari, selecciona la instal·lació i modifica les dades associades.</li> <li>4. Clica el botó de "Modificar"</li> <li>5. El sistema modifica les dades de la instal·lació.</li> <li>6. Es mostra missatge a l'usuari confirmant la modificació.</li> </ol>
Flux alternatiu	<ol style="list-style-type: none"> <li>5a. Ja existeix aquesta instal·lació i no es pot modificar.</li> <li>6a. Es mostra missatge per informar i continua en 2. del flux normal</li> </ol>
Entrada	Dades modificades de la instal·lació escollida.
Sortida	Es modifiquen les dades i continua en la pantalla de gestió d'instal·lacions.

Nom	<b>Eliminar instal·lació</b>
Descripció	Dona de baixa una instal·lació del sistema.
Actors	Gestor, Admin.
Pre condicions	L'usuari esta validat en l'aplicació.
Flux normal	<ol style="list-style-type: none"> <li>1. Selecciona l'opció d'instal·lacions del menú.</li> <li>2. Es mostra la pantalla de gestió d'instal·lacions amb el llistat d'instal·lacions.</li> <li>3. L'usuari selecciona la instal·lació i clica el boto de "Eliminar".</li> <li>4. El sistema elimina la instal·lació.</li> <li>5. Es mostra missatge confirmant la baixa.</li> </ol>
Entrada	Instal·lació seleccionada.
Sortida	S'elimina la instal·lació i continua en la pantalla de gestió d'instal·lacions.

Nom	<b>Alta de localització</b>
Descripció	Donar d'alta una nova localització (estació o trajecte).
Actors	Gestor, Admin.
Pre condicions	L'usuari esta validat en l'aplicació.
Flux normal	<ol style="list-style-type: none"> <li>1. L'usuari selecciona l'opció de localitzacions del menú.</li> <li>2. El sistema mostra la pantalla de gestió de localitzacions.</li> <li>3. L'usuari introdueix les dades de la nova estació o trajecte i fa clic en "Alta".</li> <li>4. La nova localització es dona d'alta en el sistema.</li> <li>5. Es mostra missatge confirmant l'alta.</li> </ol>
Flux alternatiu	<ol style="list-style-type: none"> <li>4a. Ja existeix la localització i no es pot donar d'alta.</li> <li>5a. Es mostra missatge per informar i continua en 2. del flux normal</li> </ol>
Entrada	Dades de la nova localització, estació o trajecte
Sortida	Es dona d'alta la localització i continua en la pantalla de gestió de localitzacions.

Nom	<b>Modificar localització</b>
Descripció	Modificar les dades d'una localització (estació o trajecte).
Actors	Gestor, Admin.
Pre condicions	L'usuari esta validat en l'aplicació.
Flux normal	<ol style="list-style-type: none"> <li>1. Selecciona l'opció de localitzacions del menú.</li> <li>2. El sistema mostra la pantalla de gestió de localitzacions amb el llistat de les estacions i trajectes existents en el sistema.</li> <li>3. L'usuari, selecciona la localització i modifica les dades associades i clica el botó de "Modificar".</li> <li>4. El sistema modifica les dades de la localització.</li> <li>5. Es mostra missatge a l'usuari confirmant la modificació.</li> </ol>
Flux alternatiu	<ol style="list-style-type: none"> <li>4a. Ja existeix aquesta localització i no es pot modificar.</li> <li>5a. Es mostra missatge per informar i continua en 2. del flux normal</li> </ol>
Entrada	Dades modificades de la instal·lació escollida.
Sortida	Es modifiquen les dades i continua en la pantalla de gestió d'instal·lacions.



Nom	<b>Eliminar localització</b>
Descripció	Donar de baixa una localització (estació o trajecte).
Actors	Gestor, Admin.
Pre condicions	L'usuari esta validat en l'aplicació.
Flux normal	<ol style="list-style-type: none"> <li>1. L'usuari selecciona l'opció de localitzacions del menú.</li> <li>2. El sistema mostra la pantalla de gestió de localitzacions amb el llistat de les estacions i trajectes existents en l'aplicació.</li> <li>3. L'usuari selecciona la localització i clica el boto de "Eliminar".</li> <li>4. El sistema dona de baixa la localització.</li> <li>5. Es mostra missatge confirmant la baixa.</li> </ol>
Entrada	Localització seleccionada.
Sortida	S'elimina la localització i continua en la pantalla de gestió de les localitzacions.

Nom	<b>Buscar continguts</b>
Descripció	El usuari pot fer una cerca de continguts d'un tipus concret.
Actors	User, Gestor i Admin.
Pre condicions	L'usuari esta validat en l'aplicació.
Flux normal	<ol style="list-style-type: none"> <li>1. L'usuari selecciona l'opció del tipus de contingut del menú.</li> <li>2. El sistema mostra la pantalla de l'opció seleccionada (vídeos, imatges o documents), i fent una cerca dels documents existents del tipus escollit.</li> <li>3. El sistema mostra un llistat amb els continguts del tipus seleccionat.</li> </ol>
Flux alternatiu	<ol style="list-style-type: none"> <li>4. L'usuari selecciona diferents criteris i clica el botó "Buscar" per fer una nova cerca més precisa en funció dels criteris.</li> <li>5. Continua en el punt 3 del flux normal.</li> </ol>
Entrada	Tipus de contingut (o vídeos o imatges o documents).
Sortida	Llistat dels continguts existents en funció del tipus escollit.

Nom	<b>Buscar continguts per localització</b>
Descripció	El usuari pot fer una cerca de continguts depenent de la localització (estació o trajecte).
Actors	User, Gestor i Admin.
Pre condicions	L'usuari esta validat en l'aplicació.
Flux normal	<ol style="list-style-type: none"> <li>1. L'usuari selecciona l'opció de continguts del menú.</li> <li>2. El sistema mostra la pantalla de continguts amb opcions per escollir la localització (estacions o trajectes) i els diferents tipus de continguts.</li> <li>3. L'usuari selecciona la localització, els diferents tipus de continguts i clica el botó "Buscar".</li> <li>4. El sistema fa una cerca dels documents existents en el sistema pertanyents a la localització concreta i dels tipus escollit.</li> <li>5. El sistema mostra a l'usuari un llistat dels continguts disponibles.</li> </ol>
Flux alternatiu	<ol style="list-style-type: none"> <li>6. L'usuari pot seleccionar diferents criteris i clica el botó "Buscar" per fer una nova cerca més precisa en funció de nous criteris.</li> <li>7. Continua en el punt 5 del flux normal.</li> </ol>
Entrada	Tipus de contingut (Al menys un tipus) i localització.
Sortida	Llistat dels continguts existents en funció de la localització.



Nom	<b>Reproduir/Mostrar continguts</b>
Descripció	Reproducció d'un vídeo, visualització d'un document, una imatge o un conjunt d'imatges
Actors	User, Gestor i Admin.
Pre condicions	L'usuari esta validat en l'aplicació i ha finalitzat el cas d'us "Buscar continguts" o "Buscar continguts per localització".
Flux normal	1. L'usuari selecciona d'un llistat, un vídeo, un document, una imatge o conjunt d'imatges. 2. El sistema reproduceix o mostra el contingut o continguts seleccionats.
Flux alternatiu 1	3. L'usuari pot tornar enrere al llistat anterior. 4. Continua en el punt 1 del flux normal
Flux alternatiu 2	2a. S'ha produït un error i no es pot reproduir/mostrar el contingut seleccionat. 3a. El sistema mostra un missatge d'error. 4a. Continua en el punt 3 del flux alternatiu 1.
Entrada	Selecció d'un vídeo, imatge, document o conjunt d'imatges.
Sortida	Visualització o reproducció dels contingut/ escollits.

Nom	<b>Alta de continguts</b>
Descripció	Donar d'alta un contingut en l'aplicació.
Actors	Gestor, Admin.
Pre condicions	L'usuari esta validat en l'aplicació i ha finalitzat el cas d'us "Buscar continguts" o "Buscar continguts per localització".
Flux normal	1. L'usuari selecciona un contingut concret o un conjunt de continguts en el cas de imatges o documents (només si pertanyen a la mateixa localització i instal·lació) i clica el botó "Alta". 2. El sistema dona d'alta el contingut i el guarda en el sistema. 3. El sistema mostra un missatge a l'usuari confirmant l'alta.
Flux alternatiu	2a. Es produeix un error i no es pot donar d'alta. 3a. Es mostra missatge per informar de l'error
Entrada	Contingut concret o un conjunt de continguts en el cas de imatges o documents (només si pertanyen a la mateixa localització i instal·lació)
Sortida	Es dona d'alta el contingut o continguts.

Nom	<b>Modificar contingut</b>
Descripció	Modificar les dades d'un contingut concret.
Actors	Gestor, Admin.
Pre condicions	L'usuari esta validat en l'aplicació i ha finalitzat el cas d'us "Buscar continguts" o "Buscar continguts per localització".
Flux normal	1. L'usuari selecciona d'un llistat, un vídeo, un document, una imatge, modifica la seva informació i clica el botó "Modificar". 2. El sistema modifica les dades associades al contingut. 3. Es mostra missatge a l'usuari confirmant la modificació.
Flux alternatiu	2a. S'ha produït un error i no es poden modificar les dades associades. 3a. Es mostra missatge per informar i continua en 1. del flux normal
Entrada	Dades modificades de un contingut.
Sortida	Llistat dels continguts existents

Nom	<b>Eliminar contingut</b>
Descripció	Eliminar del sistema un contingut concret.
Actors	Gestor, Admin.
Pre condicions	L'usuari esta validat en l'aplicació i ha finalitzat el cas d'us "Buscar continguts" o "Buscar continguts per localització".
Flux normal	1. L'usuari selecciona d'un llistat, un contingut i clica el botó "Eliminar". 2. El sistema elimina el contingut i les dades associades. 3. Es mostra missatge a l'usuari confirmant la baixa.
Flux alternatiu	2a. Es produeix un error i no es pot eliminar el contingut escollit. 3a. Es mostra missatge per informar i continua en 1. del flux normal
Entrada	Contingut a eliminar.
Sortida	Llistat dels continguts existents

### 3.3. Disseny de la base de dades.

Es segueix una metodologia en tres etapes, el disseny conceptual on es descriuen les necessitats (entitats, relacions i atributs), i es representen amb un diagrama entitat – relació, el disseny lògic, on es transforma el diagrama a un esquema relacional (amb taules) que farà servir el SGBD, i finalment el disseny físic que correspon a la implementació física de la base de dades.

#### 3.3.1. Disseny conceptual.

Les necessitats que cal tindre en compte son les següents:

Usuarios: Hem definit l'entitat "Usuarios" que contindrà les dades referents als usuaris. Els seus atributs seran, id\_usuario, nom, usuario, password i perfil. En les següent taules s'expliquen els atributs amb més claredat.

**Taula 3. Descripció dels camps de la base de dades.**

CAMP	DESCRIPCIÓ
id_usuario	Codi amb el que se farà referència a un usuari en l'aplicació.
nombre	Nom complet de l'usuari.
usuario	Codi d'identificació amb el que es valida l'usuari
password	Contrasenya de l'usuari en l'aplicació
perfil	Perfil o rol de l'usuari en l'aplicació

Instalaciones: L'entitat "Instalaciones" contindrà les dades referents a les instal·lacions ferroviàries. Els seus atributs seran, id\_instalacion, especialidad i elemento.

CAMP	DESCRIPCIÓ
id_instalacion	Codi amb el que se farà referència a una instal·lació ferroviària.
especialidad	Especialitat en la que es classifica la instal·lació.
elemento	Element de la especialitat a la que pertany la instal·lació.

Localizaciones: L'entitat "Localizaciones" contindrà les dades referents a una localització. Els seus atributs son descripció i tipo.

CAMP	DESCRIPCIÓ
id_localizacion	Codi amb el que se farà referència a una localització.
descripcion	Descripció de la localització.
tipo	Tipus de localització (estació o trajecte).

Estaciones: L'entitat "Estaciones" contindrà les dades referents a una estació. Els seus atributs son estacion i codigo.

CAMP	DESCRIPCIÓ
estación	Nom de la estació.
codigo	Codi que identifica la dependència

Trayectos: L'entitat "Trayectos" contindrà les dades referents a un trajecte entre dues estacions. Els seus atributs son origen, destino i distancia.

CAMP	DESCRIPCIÓ
origen	Nombre de la estació origen del trajecte

destino	Nombre de la estación destino del trayecto.
distancia	Distancia existente entre las estaciones origen y destí.

**Contenidos:** L'entitat "Contenidos" contindrà les dades referents als continguts, (vídeos, fotos i documents). Els seus atributs seran, id\_contenido, contenido, tipo, fecha i enlace.

CAMP	DESCRIPCIÓN
id_contenido	Codi amb el que se farà referència a un contingut.
contenido	Descripció del contingut.
tipo	Tipus de contingut, vídeo, foto o document.
fecha	Data del contingut.
enlace	Path on es troba físicament el contingut.

**Referencias:** L'entitat "Referencias" contindrà la relació entre els continguts, la seva localització i la instal·lació. Els atributs son, id\_contenido, id\_localizacion, id\_instalacion.

CAMP	DESCRIPCIÓN
id_contenido	Codi amb el que se farà referència a un contingut.
id_localizacion	Codi amb el que se farà referència a una localització.
id_instalacion	Codi amb el que se farà referència a una instal·lació ferroviària.

### 3.3.1.1. Diagrama Entitat - Relació

En la figura següent es veu com queda estructurat el diagrama entitat-relació, representant les entitats (rectangles), els diferents atributs (el·lipses) i la relació existent (rombe).

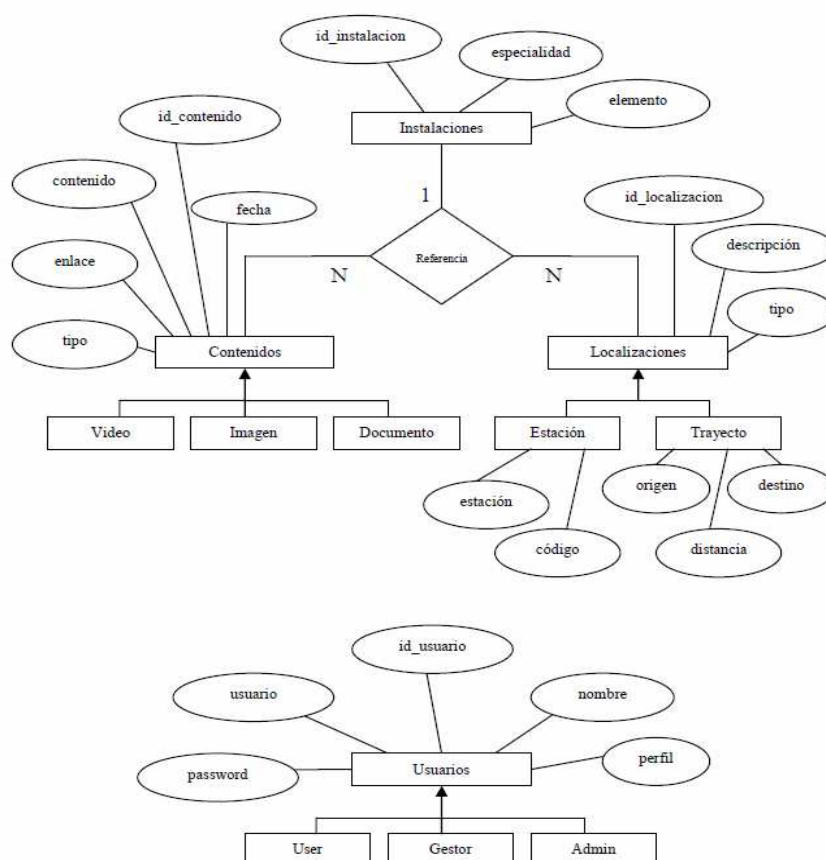


Figura 11. Esquema Entitat - Relació.

En els casos de les entitats “Contenidos”, “Usuarios” i “Localizaciones” es pot veure que es donen casos de generalitzacions, més concretament en els dos primers, es classifiquen en funció del seu tipus, però com els subtipus no tenen atributs, tal com veurem en el disseny lògic, només serà necessari un atribut en el supertipus per indicar el tipus al que pertany, ja sigui de contingut o de usuari. En el cas de les localitzacions, cadascun dels subtipus sí que tenen atributs.

La entitat “Usuarios” en principi no estableix cap relació amb les altres entitats, però s’ha representat per claredat. Aquesta entitat serà necessària per emmagatzemar la informació dels usuaris per tal de comprovar l’accés o conèixer el rol assignat.

Existeix una relació entre “Instalaciones”, “Contenidos”, “Localizaciones” que serà del tipus 1:N:N, donat que un contingut concret de un tipus d’instal·lació pot fer referència a més d’una localització, un contingut concret de una localització concreta, només farà referència a un tipus d’instal·lació, i a una localització i un tipus de instal·lacions, poden fer referència molts continguts.

### 3.3.2. Disseny lògic.

Es defineixen les taules que generalment corresponen amb cadascuna de les entitats i la relació, definides en el punt 2.3.1. En cadascuna es defineixen els valors que identifiquen de forma unívoca cada fila en cada taula, aquesta s’anomena clau primària (*PRIMARY KEY*), i no admeten valors nuls (en vermell i subratllat). Per tal de crear relacions entre taules de la base de dades (en vermell i cursiva), es faran servir claus foranies (*FOREIGN KEY*), que permeten crear una relació entre una clau primària de una taula i una forana d’una altra, per assegurar la integritat referencial de forma que al esborrar o actualitzar elements de la taula on esta definida la clau principal, s’actualitzen o esborren els elements en cascada de la taula on esta definida la clau forana.

**Taula 4. Valors dels diferents camps de la base de dades.**

Instalaciones	
CAMP	TIPUS
<u>id_instalacion</u>	int
especialidad	char (20)
elemento	char (40)

Estaciones	
CAMP	TIPUS
<u>id_localizacion</u>	int
estacion	char (50)
codigo	char (6)

Localizaciones	
CAMP	TIPUS
<u>id_localizacion</u>	int
descripcion	char (100)
tipo	enum ('estacion', 'trayecto')

Trayectos	
CAMP	TIPUS
<u>id_localizacion</u>	int
origen	char (50)
destino	char (50)
distancia	float

Contenidos	
CAMP	TIPUS
<u>id_contenido</u>	int
contenido	char (50)
fecha	Date ('yy-mm-dd')
enlace	char (80)
tipo	Enum ('video', 'foto', 'doc')

Usuarios	
CAMP	TIPUS
<u>id_usuario</u>	int
nombre	char (50)
usuario	char (20)
password	char (20)
perfil	Enum ('user', 'gestor', 'admin')

Referencias	
CAMP	TIPUS
<u>id_localizacion</u>	int
<u>id_contenido</u>	int
<u>id_instalacion</u>	int

### 3.3.3. Disseny físic.

El disseny físic correspon a la creació de les taules de la base de dades, i per tant, es comentarà posteriorment a la fase d'implementació, després de descriure en el punt 2.5. el sistema gestor de base de dades escollit.

### 3.4. Interfície d'usuari.

L'accés a l'aplicació de gestió de continguts multimèdia de l'àmbit ferroviari i el seu ús per part dels usuaris, es realitzarà per mitja d'un navegador web, i la interacció i edició de contingut es realitzarà per mitja de formularis web, en que es mostraran les dades existents per canviar-les o s'introduirà de nova. També ha de permetre veure continguts prèviament seleccionats com vídeos, imatges o documents.

El diagrama següent recull l'esquema que hauria d'oferir la interfície d'usuari de l'aplicació, i quina disposició tindrà (en la mesura que es pugui, s'intentarà seguir aquest esquema, encara que esta subjecte a posteriors decisions d'implementació).

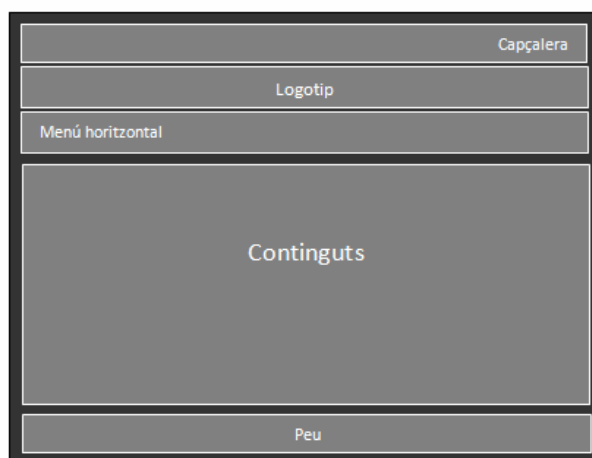


Figura 12. Esquema d'interfície d'usuari.

En la capçalera apareixeran els enllaços per tal de sortir de l'aplicació i canviar la contrasenya, en la part de logotip apareixerà una imatge. En el menú apareixeran les diferents opcions que donaran accés als diferents usos de la aplicació i dependran del perfil del tipus del usuari. Els continguts es mostraran en l'àrea que porta aquest nom, i finalment el peu es reserva per mostrar algun tipus d'informació i per millorar la presentació dels continguts en el seu conjunt.

## 4. Implementació

En la fase de implementació de l'aplicació s'instal·laran i configuraran les aplicacions escollides, es crearan les diferents taules de la base de dades (disseny físic) i es porta a terme la programació de l'aplicació per tal de garantir el compliment de les especificacions de la fase anterior.

### 4.1. Aplicacions utilitzades.

Avui en dia les tecnologies utilitzades esmentades anteriorment, s'utilitzen molt sovint en la implementació de projectes Web. Actualment existeixen eines de desenvolupament que faciliten el treball amb aquestes tecnologies com és el cas de XAMPP [12]. XAMPP disposa d'un servidor Web de software lliure sota la llicència GNU molt fàcil d'utilitzar i capaç d'interpretar les pàgines dinàmiques.

Ofereix una col·lecció de llibreries i aplicacions d'utilitat per a manegar i administrar una aplicació Web juntament amb altres dependències que resulten imprescindibles. Inclou entre altres utilitats, PEAR, OpenSSL, phpMyAdmin, Webalizer, FileZilla FTP Server, SQLite, Mercury Mail i altres, així com un panell de control específic per a XAMPP.

Per al projecte utilitzarem la versió per a Windows 1.8.2 que inclou entre altres:

- Apache 2.4.4.
- MySQL 5.5.32
- PHP 5.4.19
- phpMyAdmin 4.0.4
- OpenSSL 0.9.8
- XAMPP Control Panel 3.2.1.

Pel que respecta a la edició del codi dels arxius s'utilitzarà Notepad++ amb la versió 6.5. és tracta d'un editor de text potent amb moltes opcions que es situa entre un bloc de notes millorat i un entorn de desenvolupament lleuger. Obre els arxius en pestanyes organitzables, amb vistes alternatives per comparar text, numeració o zoom. Permet veure en diferents colors la sintaxis de molts llenguatges de programació, i que en el cas nostre projecte ens servirà per l'edició de documents CSS, JavaScript, PHP o HTML. Cal descarregar-lo prèviament de la pàgina web dels seus creadors i un cop instal·lat, ja es poden crear i editar els fitxers.

#### 4.1.1. Instal·lació i configuració de les aplicacions

En primer lloc cal procedir amb la instal·lació i configuració de les aplicacions utilitzades.

El programari XAMPP es descarrega de la pagina web dels desenvolupadors del projecte. Hi ha diferents versions, de les quals s'ha fet servir la versió amb el instal·lador de l'aplicació.

Un cop descarregat i instal·lat es pot fer servir el servidor Web Apache, PHP i el SGBD MySQL, tenint en compte que mitjançant el seu tauler de control, han de estar actius Apache i MySQL. Des del propi ordinador on s'ha fet la instal·lació s'accedeix a la URL: localhost, que coincidirà amb l'adreça IP que l'equip té assignat, de manera que aquesta serà l'adreça que farà servir Apache per mostrar el seu servei.



Figura 13. Pàgina inicial de XAMPP.

Quan s'accedeix per primer cop cal configurar-ne el idioma, i a continuació és important triar l'enllaç de "chequeo de seguretat", on es pot comprovar que per defecte l'usuari root de MySQL no té contrasenya d'accés, al igual que "phpMyAdmin" (interfície gràfica del servidor MySQL). Aquesta mateixa pàgina permet triar un enllaç per assignar les contrasenyes corresponents i solucionar-ho.

Un aspecte important és la configuració del servidor Apache per habilitar la pujada d'arxius cap al servidor amb la capacitat que es vulgui donar. Per editar la configuració del servidor, cal modificar alguns paràmetres de l'arxiu "php.ini", al qual podem accedir mitjançant el botó "Config" del XAMPP Control Panel:



Figura 14. Tauler de control de XAMPP.



Després cal canviar els tres paràmetres següents:

- `max_execution_time = 3600`, que és un paràmetre que defineix el temps màxim d'execució per cada script en segons. Passat aquest temps el servidor cancel·la la transferència, i per tant donem un temps d'una hora per a la carrega de continguts, on es important en el cas dels vídeos.
- `upload_max_filesize = 1000M`. És un paràmetre de defineix la mida màxima que PHP permet pujar, i per tant habilitem un servei de 1000M = 1 Gb.
- `post_max_size = 1000M`. És un paràmetre que defineix la mida màxima d'arxiu que el mètode POST del codi PHP. En aquest cas dona un servei màxim de 1000M = 1 Gb, valor igual que el permís de màxima mida d'arxius en PHP.

Al treballar amb XAMPP, els diferents continguts de les pàgines i fitxers han de estar col·locats sota el directori `htdocs` situat a `C:\xampp\htdocs`.

Pel nostre projecte, s'ha creat una carpeta anomenada "gestion", on penjaran tots els fitxers i continguts de l'aplicació. Concretament la estructura d'arxius i carpetes serà la següent:

- Per accedir a l'aplicació s'introduirà la següent adreça URL:  
`http://localhost/gestion` des del propi servidor o canviant "localhost" per l'adreça IP del servidor des de la Intranet.
- Dins de la carpeta gestió hi haurà l'arxiu "index.php" que servirà d'inici de l'aplicació i les següents 4 carpetes:
  - `apli` : Carpeta amb els diferents arxius php de l'aplicació.
  - `cont`: Carpeta amb els continguts (vídeos, imatges i documents).
  - `estils`: Carpeta amb arxius de fulls d'estils i imatges de disseny
  - `js`: Carpeta amb llibreries javascript com per exemple jQuery.

## 4.2. Formats dels fitxers dels continguts

Degut a la limitació existents en els diferents formats al fer ús de la etiqueta `<video>` de HTML5, els únics formats suportats per aquest estàndard son `mp4`, `ogg` i `webm`, i per tant les úniques extensions admeses per als vídeos seran `.mp4`, `.ogg` i `.webm` respectivament, respecte a les imatges, els formats seran `.jpg`, `.jpeg`, i pel que fa als documents inicialment només es contempla `.pdf`.

Cal tenir en compte que no tots els navegadors suporten tots els formats, i a més, els resultats poden ser diferents depenent de les successives versions dels diferents navegadors.

Provant els navegadors més populars, tenim que:

- Internet Explorer: suporta la etiqueta “video” des de la versió 9 i l’únic format que admet es mp4 des d’aquesta versió.
- Mozilla Firefox: suporta la etiqueta video des dels inicis (versió 3.5) i els formats admesos sempre han estat ogv i webm. En les versions 24 i 25 també admetien mp4 però en les següents no.
- Google Chrome: suporta la etiqueta video des de la versió 3, i a partir de la versió 6 suporta els tres formats de vídeo de HTML5.

Es pot dir que si tindríem que escollir-ne un, el navegador més adient seria Google Chrome donat que suporta els 3 formats, seguit de Mozilla Firefox que en suporta 2 i Internet Explorer que en suporta només 1.

Respecte als formats de video, Mozilla en principi no admet mp4, però normalment amb la instal·lació del navegador, porta per defecte un plugin Quick Time que permet visualitzar-los. Donat que normalment és el format més popular, i que Internet Explorer i Google Chrome si el suporten per defecte, és recomanarà ara per ara que aquest sigui el format proposat per als vídeos de l’aplicació. En qualsevol cas, com tot depèn finalment de la combinació entre navegador Web i format de video qualsevol dels 3 s’admet.

Evidentment comprovar tant la correcta visualització dels vídeos en els diferents formats i el correcte funcionament en l’ús dels formularis en totes les versions dels navegadors, seria una tasca més que impossible, però al menys en la següent taula s’indiquen amb fons gris les versions a provar (la de Internet Explorer 8 i la última dels 3) i la compatibilitat de les últimes versions dels navegadors i l’ús dels 3 formats i l’etiqueta vídeo.

	video	mp4	ogv	webm
Internet Explorer 8	No	No	No	No
Internet Explorer 9	Si	Si	No	No
Internet Explorer 10	Si	Si	No	No
Internet Explorer 11	Si	Si	No	No
Mozilla Firefox 24	Si	Si	Si	Si
Mozilla Firefox 25	Si	Si	Si	Si
Mozilla Firefox 26	Si	No	Si	Si
Google Chrome 26	Si	Si	No	No
Google Chrome 30	Si	Si	Si	Si
Google Chrome 31	Si	Si	Si	Si

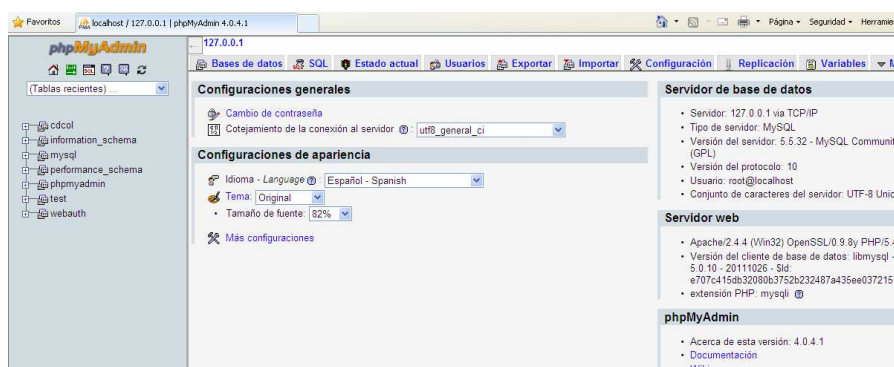
**Taula 5. Compatibilitat navegadors i formats HTML5.**

### 4.3. Disseny físic de la base de dades.

Aquesta fase consisteix en la creació de les taules de la base de dades del projecte. Tal com s'ha esmentat en el punt 3.1.1., al introduir l'adreça IP o "localhost" en la barra de direcció del navegador s'accedeix a la pàgina principal de XAMPP.

Per al projecte caldrà crear una base de dades que anomenarem "gestion" amb les diferents taules, i d'altra banda serà necessari donar d'alta un usuari amb tots els permisos, de manera que quan s'accedeixi des del codi PHP no hi hagi cap problema.

Al costat esquerre de la pagina principal de XAMPP hi ha disponible un enllaç a la interfície gràfica "phpMyAdmin" del servidor MySQL, tal i com es veu a la següent figura:



**Figura 15. Interfície gràfica de phpMyAdmin.**

Des de la interfície gràfica cal donar d'alta un nou usuari seleccionant la pestanya "Usuarios" de la part superior de la figura anterior assignant-li usuari, contrasenya i els privilegis adients per tal de que posteriorment des del codi de php es pugui connectar amb la base de dades i poder fer les consultes i operacions que calguin.

També tal tenir en compte especificar el conjunt de caràcters que s'utilitzarà. En el nostre cas farem servir UTF-8 i és important especificar-ho tant en les pàgines web, com en la base de dades i les taules per a que no hi hagin problemes (per exemple amb la accentuació).

Finalment, cal tenir en compte que el motor de emmagatzemament per defecte en aquesta versió de MySQL és InnoDB (tecnologia de emmagatzemament de dades caracteritzada per suportar transaccions amb integritat referencial). Aquesta integritat referencial garanteix que una entitat es relaciona amb una altra que existeix en la base de dades, assegurant en tot moment que la informació existeixi, que no esta repetida innecessariament, etc. InnoDB té la particularitat de permetre definir claus forànies i per tant, podem definir regles o restriccions que asseguraran la integritat referencial tal com s'havia vist en la fase de disseny lògic (punt 3.3.2).

Per procedir a crear la base de dades només cal seleccionar la pestanya de “Base de datos” de la figura anterior i introduir el nom (gestion en el nostre cas) i triar el “cotejamiento” amb UTF8\_general\_ci. Un cop creada ja podem afegir les diferents taules. Per cadascuna en primer lloc cal indicar el nom i el nombre de files, i en la següent finestra s’introdueixen els diferents camps, tipus, si es null, etc.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra	Acción
1	id_usuario	int(11)	utf8_general_ci		No	Ninguna	AUTO_INCREMENT	Cambiar, Eliminar, Primaria
2	nombre	varchar(50)	utf8_general_ci		No	Ninguna		Cambiar, Eliminar, Primaria
3	usuario	varchar(20)	utf8_general_ci		No	Ninguna		Cambiar, Eliminar, Primaria
4	password	varchar(20)	utf8_general_ci		No	Ninguna		Cambiar, Eliminar, Primaria
5	perfil	enum('user', 'admin', 'gestor')	utf8_general_ci		No	Ninguna		Cambiar, Eliminar, Primaria

Figura 16. Vista de la estructura de la taula "usuarios".

Tabla	Acción	Filas	Tipo	Cotejamiento	Tamaño
contenidos	Examinar, Estructura, Buscar, Insertar, Vaciar, Eliminar	~2	InnoDB	utf8_general_ci	1.6 KB
estaciones	Examinar, Estructura, Buscar, Insertar, Vaciar, Eliminar	~2	InnoDB	utf8_general_ci	1.6 KB
instalaciones	Examinar, Estructura, Buscar, Insertar, Vaciar, Eliminar	~5	InnoDB	utf8_general_ci	1.6 KB
localizaciones	Examinar, Estructura, Buscar, Insertar, Vaciar, Eliminar	~3	InnoDB	utf8_general_ci	1.6 KB
referencias	Examinar, Estructura, Buscar, Insertar, Vaciar, Eliminar	~2	InnoDB	utf8_general_ci	4.8 KB
trayectos	Examinar, Estructura, Buscar, Insertar, Vaciar, Eliminar	~1	InnoDB	utf8_general_ci	1.6 KB
usuarios	Examinar, Estructura, Buscar, Insertar, Vaciar, Eliminar	~3	InnoDB	utf8_general_ci	1.6 KB
7 tablas	Número de filas	18	InnoDB	utf8_general_ci	144 KB

Figura 17. Vista de les taules amb la interfície phpMyAdmin.

Des de la interfície gràfica resulta senzilla la creació i configuració de les diferents taules, però evidentment té la seva traducció amb sentències SQL. També es poden crear les taules introduint directament les sentències amb SQL des de la pestanya que porta aquest nom.

Per exemple la sentència per crear aquesta taula amb SQL seria:

```
CREATE TABLE 'usuarios' (
  'id_usuario' int(11) unsigned NOT NULL auto_increment,
  'nombre' varchar(50) NOT NULL,
  'usuario' varchar(20) NOT NULL,
  'password' varchar(20) NOT NULL,
  'perfil' enum('user', 'admin', 'gestor') NOT NULL,
  PRIMARY KEY ('id_usuario')
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=1;
```

Cal tenir en compte que les taules “estaciones”, “trayectos” i “referencias”, tenen claus forànies, i per tant cal indicar-ho modificant-lo en la vista de relacions de cada taula o

amb una sentència SQL. Per exemple, per afegir la clau forana en la taula de “estaciones” seria:

```
ALTER TABLE 'estaciones' ADD FOREIGN KEY ('id_localizacion') REFERENCES  
'gestion'.localizaciones('id_localizacion) ON DELETE CASCADE;
```

Això permetrà que quan s'elimina una localització de la taula de localitzacions amb la seva clau primària “id\_localizacion”, s'esborren automàticament de la taula d'estacions si existeix un registre amb aquest identificador.

## 4.4. Implementació de la lògica i la presentació.

Un cop implementada la base de dades cal procedir amb la part de la lògica i la presentació. Tota la part lògica es desenvolupa mitjançant scripts continguts en els fitxers PHP, en els que podrem trobar tant el propi codi del costat del servidor, com llenguatge HTML, i codi en JavaScript. Pel que fa a la part de la presentació els diferents navegadors interpretaran el codi HTML i el de JavaScript enviat pel servidor.

En aquest apartat apareixeran algunes parts de codi per intentar explicar com s'implementen les funcionalitats o com exemples. Per major informació, en els diferents arxius de PHP es pot trobar molt més documentada l'explicació més detallada amb el funcionament que ha de servir per major comprensió i futures modificacions de codi.

### 4.4.1. Implementació de la lògica de l'aplicació.

L'aplicació fa servir sessions d'usuari i variables de sessió. Les sessions permeten a l'usuari navegar per les pàgines de l'aplicació amb un sol identificador d'usuari mentre hi sigui en l'aplicació i no surti de ella, permetent identificar-lo, i les variables de sessió permeten emmagatzemar informació que forma part d'una sessió concreta.

Tal com s'esmentava en l'apartat 3.1.1.2., per accedir a l'aplicació s'introduirà l'adreça URL `http://localhost/gestion` des del propi servidor o canviant “localhost” per l'adreça IP del servidor des de la Intranet i obrirà el fitxer “index.php” que està dintre del directori “gestion” que penja de `c:\xampp\htdocs`.

Aquest fitxer que dona accés a l'aplicació, presenta una part corresponent al inici de sessió, i una altra amb el codi HTML en la que hi han la capçalera, una imatge, el menú, el contingut i un peu de pàgina. Concretament 3 de elles, la capçalera, el menú i el contingut inclouen (mitjançant la directiva *include* de php), 3 fitxers PHP respectivament, “capçalera.php”, “menu.php” i “cont.php”, que mostraran diferent informació en funció del tipus d'usuari i de la funcionalitat de que es tracte.

#### 4.4.1.1. Connexió amb la base de dades.

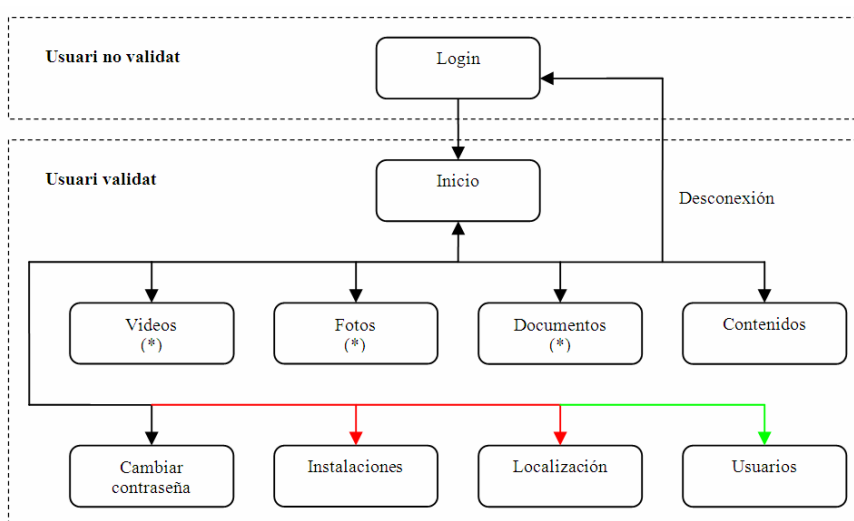
Per fer consultes a la BBDD, s'ha creat un fitxer anomenat "conexionBBDD.php" que s'inclourà mitjançant la directiva *include*, en els diferents fitxers PHP que hagin de fer alguna sentència de manipulació de dades. En aquest fitxer es realitza la connexió i selecció de la base de dades amb els paràmetres host, base de dades (gestion) i usuari i contrasenya del usuari amb privilegis que es va crear en l'apartat 4.2.

#### 4.4.1.2. Navegació.

En la capçalera ("capçalera.php"), un cop validats apareixeran 2 enllaços que permetran a qualsevol usuari canviar la contrasenya i sortir de l'aplicació, depenent de una variable de sessió (user\_login).

En la part de menú ("menu.php"), un cop validat l'usuari es mostraran els diferents enllaços per mostrar i gestionar els continguts. Els enllaços visualitzats dependran del tipus o rol de l'usuari validat i es controlen amb una variable de sessió (perfil). Concretament tots els usuaris tindran accés als enllaços de "videos", "fotos", "documentos" i "contenidos", els gestors i administradors, a més tindran accés als enllaços de "instalaciones" i "localizaciones", i finalment els administradors tindran accés també a l'enllaç de "usuarios" per a la seva gestió.

Quan es selecciona un dels enllaços (tant del menú com de la capçalera), es crida al fitxer auxiliar "accions.php" amb un paràmetre opcional (op) amb el mètode GET que conté un valor que s'utilitzarà per indicar amb una variable de sessió (contenido) quina es la part que es deu d'actualitzar en la part de continguts. Finalment es redirecciona cap a la pàgina d'inici.



(\*) Accessibles per tots els usuaris, però algunes funcionalitats només les poden fer els gestors i administradors.

■ Tots els usuaris. ■ Gestors i administradors, ■ Només administradors.

**Figura 18. Esquema de navegació de l'aplicació en funció del tipus d'usuari**



Finalment en la part de continguts (“cont.php”) es mostrarà tota la informació en funció de l’enllaç seleccionat en la part del menú o el formulari per canviar la contrasenya (simplement mirant el valor de la variable de sessió (contenido). Aquesta és la part fonamental de l’aplicació, en la que s’han dividit les diferents funcionalitats en parts més petites per anar implementant-les de forma progressiva facilitant alhora la depuració de possibles errors.

Cadascuna de les funcionalitats en que es divideix l’aplicació, estarà implementada amb 1 o més fitxers PHP, en el que el seu nom identifica la part amb la que està relacionada i/o la funcionalitat que implementa, per exemple, tots els fitxers que en un moment donat s’utilitzen en la part de continguts, comencen amb nom “cont\_” i van seguits de un nom que referència la funcionalitat que implementen. Tindrem cont\_video (videos), cont\_fotos (fotografies), cont\_docs (documents), cont\_contenidos (continguts de qualsevol tipus en funció de la localització), cont\_localizacion (estacions i trajectes), cont\_instal (instalacions), cont\_login (accés a l’aplicació), i cont\_modpassw (modificació de contrasenya).

Els fitxers que intervenen en cadascuna de les funcionalitats es comenten amb més detall en els següents apartats.

#### 4.4.1.3. Accés a l’aplicació.

Usuaris: Tots. Fitxers (2): “cont\_login\_form.php” i “cont\_login\_valida\_user”.

Per accedir a l’aplicació cal introduir el nom del usuari i la contrasenya en el formulari que apareix en l’àrea de contingut (implementat en el fitxer “cont\_login\_form.php”). Un cop introduïdes les dades s’envia mitjançant el mètode POST al fitxer “cont\_login\_valida\_user” fent la següent consulta a la base de dades per fer la comprovació de les dades:

```
$q_login = "select * from usuarios where user=('" . $_POST["username"] . "') and pass=('" . $_POST["password"] . "')";  
$rs_login=mysql_query($q_login,$conn);  
$res_login=mysql_fetch_array($rs_login,MYSQL_ASSOC);
```

Un cop executada la consulta, si ens retorna les dades del usuari, ens indica que són correctes. Llavors s’inicialitzen altres variables de sessió que controlaran entre altres que l’usuari està validat, el perfil assignat, el contingut següent a visualitzar o el nom d’usuari, i en cas contrari la informació del error en les dades. Finalment és crida novament a la pàgina d’inici, on ara en funció de si han estat validades o no les credencials, mostrarà o la pàgina d’inici amb els enllaços del menú activats en funció del perfil, o es mostrarà novament el formulari de login.



Figura 19. Pantalla de login

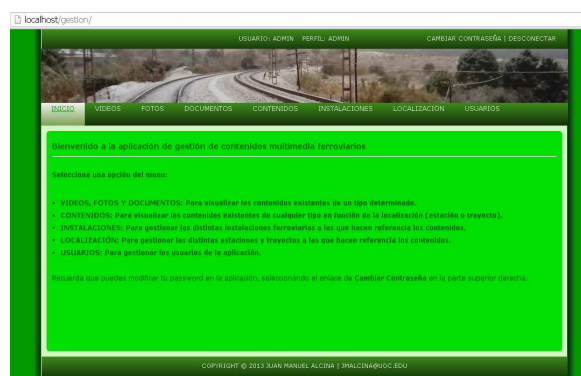


Figura 20. Pantalla inicial després de validat

#### 4.4.1.4. Desconnexió o sortida de l'aplicació.

Usuaris: Tots. Fitxers (1): “desconexión.php”.

Qualsevol usuari validat pot seleccionar l'enllaç de “desconectar” disponible en la capçalera. Al fer clic, es crida al fitxer “desconexion.php”, on s'eliminen les variables de sessió, s'elimina la cookie de sessió creada pel navegador de l'usuari, s'elimina la sessió i finalment es carrega la pagina amb el formulari de login.

#### 4.4.1.5. Canvi de contrasenya.

Usuaris: Tots. Fitxers (2): “cont\_modpassw\_form.php”, i “cont\_modpassw\_valida\_passw.php”.

Qualsevol usuari validat té l'opció de modificar la contrasenya. Per fer-ho, pot seleccionar l'enllaç de “cambiar contraseña” disponible en la capçalera (part superior dreta). Al fer clic, es carregarà a la part de continguts el fitxer “cont\_modpassw\_form.php”, presentant per pantalla un formulari per introduir la contrasenya actual i la nova dues vegades. Quan l'usuari introdueix les dades, es crida a una funció en javascript inclosa en el fitxer que comprova que els camps no estan buits i que la contrasenya nova introduïda dues vegades es la mateixa. De ser correcte s'envia el formulari al fitxer “cont\_modpassw\_valida\_passw.php”, que s'encarregarà de fer la consulta de les dades del usuari i en cas de ser correctes s'actualitza la contrasenya. Finalment es carrega la pagina d'inici.

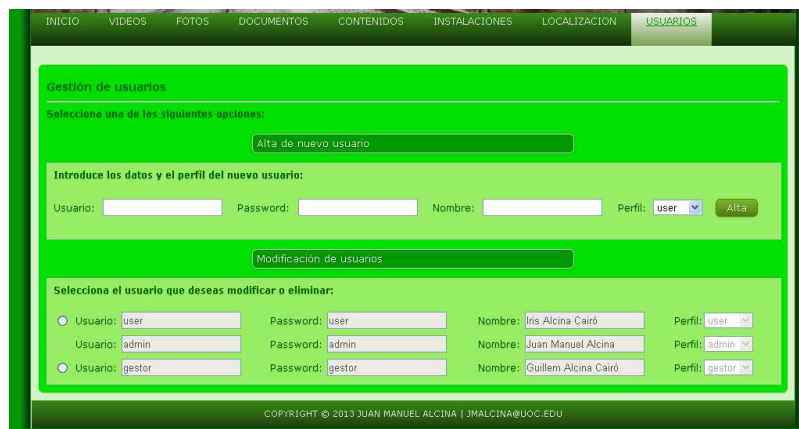
Figura 21. Vista del formulari de canvi de contrasenya.



#### 4.4.1.6. Gestió d'usuaris.

Usuaris: Admin. Fitxers (3): “cont\_usuaris\_form.php”, “cont\_usuaris\_alta\_user.php”, i “cont\_usuaris\_mod\_user.php”.

Els administradors tenen accés a l'enllaç disponible en el menú. Al fer clic, es carregarà a la part de continguts el fitxer “cont\_usuaris\_form.php”, presentant 2 panells desplegable que permeten obrir dos formularis diferents, un per donar d'alta un nou usuari, i un altre que presenta un llistat de tots els usuaris existents.



**Figura 22. Vista de la pantalla d'usuaris amb els 2 panells desplegats**

##### 4.4.1.6.1. Alta d'usuaris.

Per donar d'alta un usuari cal completar els camps de text disponibles per introduir les dades d'usuari, contrasenya, nom i el perfil que tindrà. Un cop introduïdes les dades al seleccionar el boto d'alta es crida a una funció javascript que comprovarà que s'han omplert tots els camps. De ser correcte s'envia les dades del formulari mitjançant el mètode POST al fitxer “cont\_usuaris\_alta\_user.php”. En aquest fitxer inicialment es fa una consulta a la base de dades per comprovar que no hi ha cap més que tingui aquest usuari, i en aquest cas l'insertarà a la base de dades:

```
$q_alta = "INSERT INTO usuarios (user, pass, nom, perfil) VALUES
('".$_POST["username"]."', '".$_POST["password"]."', '".$_POST["nom"]."',
 '".$_POST["perfil"]."');
$r_alta = mysql_query($q_alta, $conn);
```

En cas d'existir es mostra un missatge indicant-ho i finalment es carrega novament la pagina anterior tant si s'ha afegit com si no.

##### 4.4.1.6.2. Baixa i modificació d'usuaris

En el segon formulari es presenta el llistat dels usuaris existents amb els seus camps de text que no es poden editar. Cal seleccionar un d'ells fent clic sobre el boto radio de l'usuari que es vulgui modificar o eliminar. Un cop seleccionat un d'ells si es selecciona el botó eliminar es crida a una funció javascript que ho comprovarà, i de ser correcte

envia les dades del formulari mitjançant el mètode POST al fitxer “cont\_usuaris\_mod\_user.php” amb un camp de text ocult (hidden), que indicarà l’acció que es vol fer (en aquest cas eliminar), i el identificador de l’usuari. Es reben els paràmetres, en el fitxer i si l’acció és “eliminar”, s’esborra l’usuari de la base de dades:

```
if ($accio == "eliminar") {
    $q_delete = "DELETE FROM usuarios WHERE idusuario=$id";
    $r_delete = mysql_query($q_delete, $conn); }
```

Modificación de usuarios					
Selecciona el usuario que deseas modificar o eliminar:					
<input checked="" type="radio"/>	Usuario: user	Password: user	Nombre: Iris Alcina Cairó	Perfil: user	<input type="button" value="Modificar"/> <input type="button" value="Eliminar"/>
<input type="radio"/>	Usuario: admin	Password: admin	Nombre: Juan Manuel Alcina	Perfil: admin	
<input type="radio"/>	Usuario: gestor	Password: gestor	Nombre: Guillem Alcina Cairó	Perfil: gestor	

**Figura 23. Vista del formulari per modificar o eliminar un usuari.**

El procés de modificació es semblant al de baixa. Al triar un usuari, es poden modificar les dades corresponents a aquest i finalment cal seleccionar el botó “Modificar” on es crida a una funció javascript que comprovarà que tots els camps de l’usuari contenen dades i de ser correcte s’envien les dades del formulari mitjançant el mètode POST al fitxer “cont\_usuaris\_mod\_user.php” amb un camp de text ocult (hidden), que indicarà l’acció que es vol fer (en aquest cas modificar), i el identificador de l’usuari. Es reben els paràmetres, en el fitxer i es fa una consulta del usuari, tenint en compte que no se li pot assignar el nom d’un altre, i de ser així es modificaran les dades. Finalment, tant quan es fa una baixa com una modificació retorna a la pagina inicial d’usuaris.

#### 4.4.1.7. Gestió de localitzacions.

Usuaris: Gestor i Admin. Fitxers (5): “cont\_localizacion\_form.php”, “cont\_localizacion\_alta\_estacion.php”, cont\_localizacion\_mod\_estacion.php, cont\_localizacion\_alta\_trayecto.php”, i “cont\_localizacion\_mod\_trayecto.php”.

Els gestors i administradors tenen accés a l’enllaç disponible en el menú de “localización”. Al fer clic, es carregarà a la part de continguts el fitxer “cont\_localizacion\_form.php”, presentant 2 panells desplegable corresponents a “Estación” o “Trayecto entre estaciones”. Seleccionant el panell d’estació es mostren 2 formularis, un per introduir les dades d’una estació i donar-la d’alta i un altre on apareix un llistat amb les estacions existents, en les que es pot triar una d’elles i modificar les seves dades o eliminar-la.

El procés és idèntic respecte al panell de trajectes.

**Figura 24. Selecció d'una estació per modificar les dades.**

En funció de la acció que es porta a terme (quin dels 4 formularis és selecciona), ens portarà cap a un dels altres 4 fitxers, seguint el mateix procés que en la gestió d'usuaris, excepte que en aquest cas després de donar de alta les dades corresponents a la localització, es necessari obtenir el identificador de la localització (`id_localizacion`), mitjançant la funció "`mysql_insert_id()`" i fer una nova inserció de les dades de la estació o trajecte. El següent codi mostra per exemple les accions que es duen a terme en el fitxer de alta d'una estació, que correspon logicament a "`cont_localizacion_alta_estacion.php`":

```
$q_alta_loc = "INSERT INTO localizaciones (descripcion, tipo) VALUES ('".$_POST["descripcion"]."', 'estacion)";
$r_alta_loc = mysql_query($q_alta_loc, $conn);
$id = mysql_insert_id();
$q_alta_est = "INSERT INTO estaciones (id_localizacion, estacion, codigo) VALUES ('".$id."', '".$_POST["estacion"]."', '".$_POST["codigo"]."')";
$r_alta_est = mysql_query($q_alta_est, $conn);
```

En qualsevol cas després de dur a terme l'alta, baixa o modificació, tant de una estació, com d'un trajecte es carregarà la pagina inicial de localitzacions.

#### 4.4.1.8. Gestió d'instal·lacions.

Usuaris: Gestor i Admin. Fitxers (3): "`cont_instal_form.php`", "`cont_instal_alta_elemento.php`", i "`cont_instal_mod_elemento.php`".

Cada contingut fa referència a algun tipus d'instal·lació, formada per una especialitat i per un element. Seleccionant per part dels usuaris o administradors l'enllaç disponible en el menú de "instalaciones", es carregarà a la part de continguts el fitxer "`cont_instal_form.php`", presentant 2 panells desplegable que permeten obrir dos formularis diferents, un per donar d'alta una instal·lació, i un altre que presenta un llistat de tots els instal·lacions existents. Les especialitats estan definides en les especificacions, i es pot crear qualsevol element dintre de cada especialitat. La

funcionalitat és totalment idèntica a la d'usuaris, permetent donar d'alta, baixa i modificar, i només canvia la taula de la base de dades (instalaciones), i que els camps del formulari son “especialidad” i “elemento”.

**Figura 25. Vista dels formularis de gestió de instal·lacions**

#### 4.4.1.9. Gestió de vídeos.

Usuaris: Tots (reproducció) i Gestor i Admin (alta, baixa i modificació).

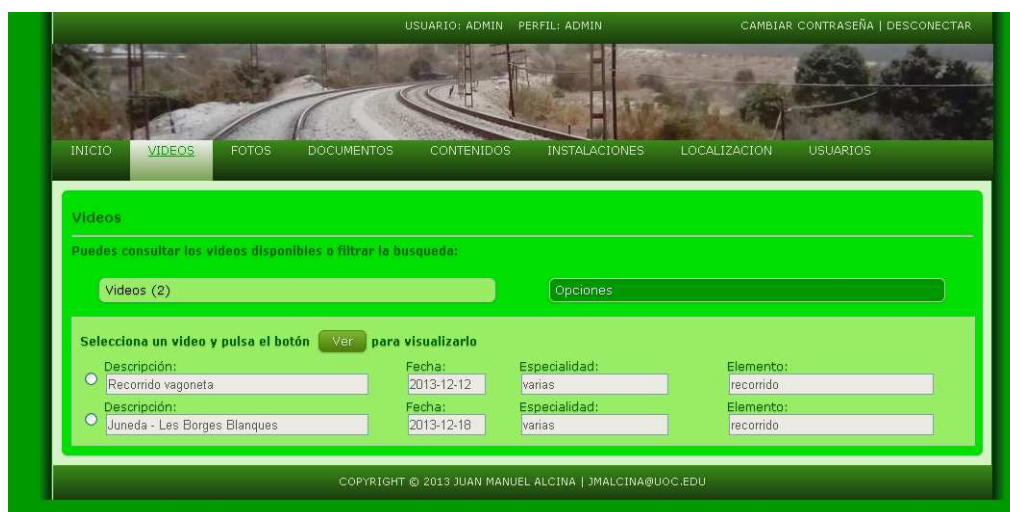
Fitxers (5): “cont\_video.php”, “cont\_video\_form.php”, “cont\_video\_ver.php”, “cont\_video\_alta.php” i “cont\_video\_mod\_video.php”.

Si qualsevol usuari selecciona l'enllaç de “videos” del menú, es carregarà a la part de continguts el fitxer “cont\_video.php”. En aquest fitxer amb una directiva *include*, inclourà algun dels altres en funció de si esta definida una variable del mètode POST anomenada “opcion” i del seu valor, concretament, si no esta definida aquesta variable s'inclou el fitxer “cont\_video\_form.php”, en el cas de estar inclourà el fitxer “cont\_video\_ver”, i si la variable té valor “alta”, inclou el fitxer “cont\_video\_alta.php”. Finalment l'altre fitxer anomenat “cont\_video\_mod\_video.php” es crida mitjançant el pas d'un formulari des del fitxer “cont\_video\_form.php” quan es vol modificar o eliminar un vídeo.

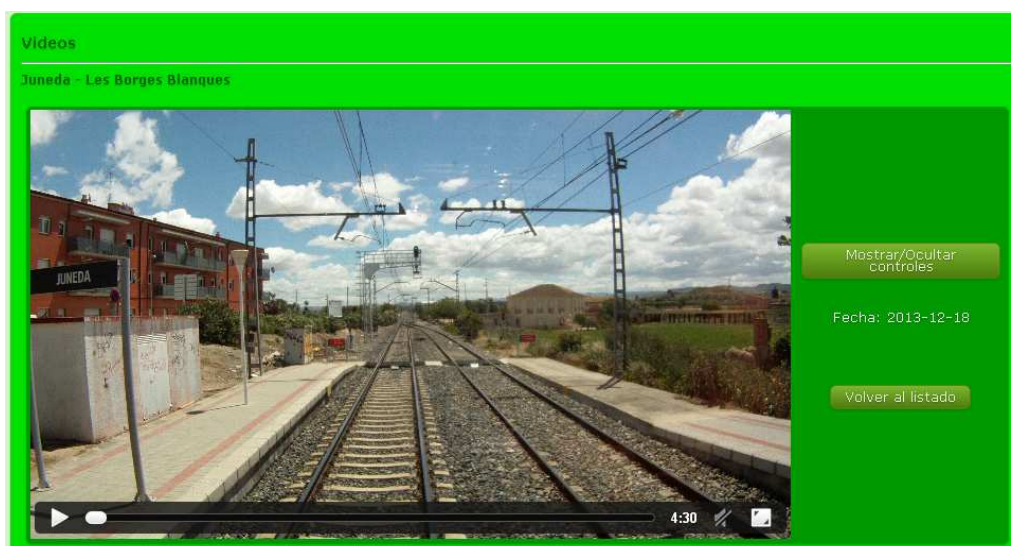
##### 4.4.1.9.1. Llista de vídeos i opcions.

El contingut que es mostra inicialment al accedir als vídeos, es el del fitxer “cont\_video\_form.php”, que presenta 2 panells. Al desplegar el panell esquerre mostra el llistat dels vídeos disponibles amb opció de seleccionar-ne un d'ells mitjançant un element input de tipus radio de formulari. Si es marca un d'ells i es fa clic en el botó “Ver”, s'envia el formulari carregant el fitxer “cont\_video\_ver.php” que permetrà

reproduir el vídeo seleccionat, passant-li en el formulari el identificador del contingut (id\_contenido).



**Figura 26. Vista del panell esquerre desplegat amb la llista de vídeos.**



**Figura 27. Detall de la pantalla per visualitzar un vídeo.**

En el fitxer “cont\_video\_ver.php” es fa una cerca en la base de dades del contingut per obtenir l'enllaç al fitxer físic del vídeo i es mostrarà per pantalla amb l'opció de reproduir-lo, sempre que el format del fitxer de vídeo sigui compatible amb el navegador de l'usuari, tenint en compte que fem us de HTML5, i que per tant els únics formats reproduïbles son mp4, webm i ogv.

El panell dret (“opciones”) del fitxer cont\_video\_form.php” presenta 3 formularis. El primer d'ells esta disponible per tots els usuaris i permet fer un filtrat del llistat dels vídeos que apareixen en l'altre panell en funció de la especialitat de la instal·lació i de la localització. Un cop triat el filtre y seleccionat el botó “Filtrar” es carrega novament la

pàgina però amb els vídeos que compleixen la selecció amb un missatge que indica que el llistat presentat es el resultant després d'aplicar un filtrat. En cas de tractar-se d'un gestor o administrador, visualitzarà els altres dos panells que al desplegar-los mostren els altres 2 formularis (un per alta de vídeo, i l'altre per modificar les seves dades o eliminar-lo).

**Figura 28. Vista de les opcions de filtrat i dels formularis d'alta, i modificació.**

En el formulari d'alta caldrà introduir les dades, seleccionar una localització i una instal·lació a la que fa referència, i seleccionar el fitxer amb un element input de tipus file de formulari. Si es selecciona el botó d'alta s'enviaran les dades del formulari cap al fitxer "cont\_video\_alta.php" i es pujarà el vídeo. Les accions que es duen a terme en aquest fitxer són en primer lloc les de inserir (1) en les taules "contenidos" i "referencias" les dades del nou contingut en la que inicialment el camp "enlace" pren el valor "temporal", a continuació (2) es verifica mitjançant la funció "is\_upload\_file()" que el fitxer s'ha pujat correctament al fitxer temporal emmagatzemat a la variable "\$\_FILES" i de ser així, es copia (3) al directori "cont" on estaran tots els continguts de l'aplicació, i finalment s'actualitza el contingut (4) modificant la informació del contingut, modificant el valor del camp "enlace" amb el identificador del contingut i la extensió del fitxer original. En el cas de que hi hagi algun problema en la carrega, s'elimina (5) el contingut de la base de dades:

- ```
(1) $q_alta_video = "INSERT INTO contenidos (contenido, fecha, enlace,tipo) VALUES ('".$_POST["descripcion"]."', '".$_POST["fecha"]."', 'temporal', 'video')";
    $r_alta_video = mysql_query($q_alta_video, $conn);

    $id= mysql_insert_id();

    $q_alta_ref = "INSERT INTO referencias (id_localizacion, id_contenido, id_instalacion)
    VALUES ('".$_POST["localizacion"]."', ".$id."', '".$_POST["instalacion"].'")";
```



```

$r_alta_ref = mysql_query($q_alta_ref, $conn);

(2)  if (is_uploaded_file($_FILES['fichero']['tmp_name']))
    {
        $archivo_nombre = $_FILES['fichero']['name'];
        $extension= explode(".", $archivo_nombre);
        $archivo_temporal = $_FILES['fichero']['tmp_name'];
        $destino= "cont/".$id.". ".$extension[1];

(3)      if (move_uploaded_file($archivo_temporal, $destino)) {
(4)          $q_update_video = "UPDATE contenidos SET
            enlace='".$id."'.".$extension[1]." WHERE id_contenido=$id";
            $r_update_video = mysql_query($q_update_video, $conn);

            header("Location: apli/accions.php?op=videos");
        }
        else { $errorcarga="error"; }
    }
    else { $errorcarga="error"; }
(5)  if ($errorcarga=="error") {
        $q_delete = "DELETE FROM contenidos WHERE id_contenido=$id";
        $r_delete = mysql_query($q_delete, $conn); }

```

Finalment, en el tercer formulari apareixen les dades del vídeo que està seleccionat en el panell del llistat (sempre que només hi hagi un sol seleccionat), permetent modificar els seus camps i seleccionar el boto corresponent a “Modificar” o “Eliminar”. En aquest cas es passa la informació del formulari al fitxer “cont\_video\_mod\_video.php” que funciona al igual que la resta de funcionalitats de modificació o baixa, exceptuant que en aquest últim cas, cal també esborrar físicament el fitxer físic del vídeo que es farà amb la funció “unlink(\$ruta)”, on “\$ruta” conté l’adreça de l’arxiu al servidor.

#### 4.4.1.10. Gestió de fotos.

Usuaris: Tots (visualització) i Gestor i Admin (alta, baixa i modificació).

Fitxers (5): “cont\_fotos.php”, “cont\_fotos\_form.php”, “cont\_fotos\_ver.php”, “cont\_fotos\_alta.php” i “cont\_fotos\_mod\_foto.php”.

La funcionalitat en la gestió de les fotos es la mateixa que la dels vídeos, excepte que en el llistat de les fotos, cadascuna disposa de un element input de tipus checkbox de formulari per permetre seleccionar-ne més d’una. Si es fa clic en el botó “Ver”, s’envia el formulari amb els identificadors de les diferents fotografies, de manera que es podran veure totes al carregar-se el fitxer “cont\_fotos\_ver.php”. En aquest fitxer és fa una crida a un visor de fotografies de codi obert anomenat GalleryView [13] desenvolupat per Jack Wanders que utilitza les llibreries de javascript jQuery [14] i jQuery UI [15], i que permet veure les imatges de forma seqüencial o seleccionant-les específicament. També té l’opció de mostrar la informació associada a cadascuna de les fotografies.

L'altra diferència amb la gestió de vídeos, és que des del formulari d'alta es pot seleccionar més d'un arxiu, evidentment sempre que el conjunt de fotos que es pugen al servidor pertanyen a la mateixa localització i mateixa instal·lació. Aquesta funcionalitat s'implementa gràcies a l'especificació que permet HTML5, de poder definir el atribut "multiple" dintre del *input* de formulari de tipus *file*. En el capítol 5 de experiments es comenta amb major detall.



Figura 29. Detall del llistat de fotos disponibles i seleccionades.

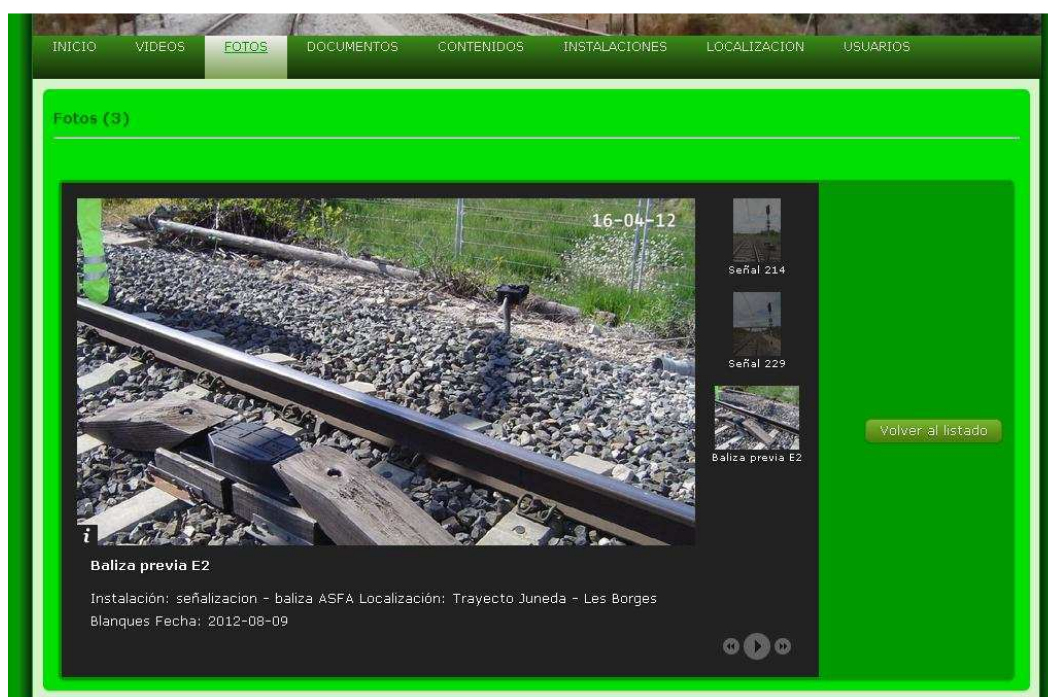


Figura 30. Visualització de les fotografies seleccionades.

#### 4.4.1.11. Gestió de documents.

Usuaris: Tots (visualització) i Gestor i Admin (alta, baixa i modificació).

Fitxers (4): "cont\_docs.php", "cont\_docs\_form.php", "cont\_docs\_alta.php" i "cont\_docs\_mod\_doc.php".

La funcionalitat en la gestió de documents és semblant a la dels vídeos, excepte que en aquest cas no cal un fitxer per mostrar el contingut. Concretament, des del llistat de documents mostrat al carregar el fitxer "cont\_docs\_form.php" només cal seleccionar el



botó “Ver” corresponent al document que es vol veure, i s’obrirà directament. En aquest cas, al igual que amb l’opció de les fotos, també és poden pujar un conjunt d’arxius cap al servidor sempre que pertanyen a la mateixa localització i mateixa instal·lació.

#### 4.4.1.12. Continguts

Usuaris: Tots. Fitxers (8): “cont\_contenidos.php”, “cont\_contenidos\_form.php”, “cont\_contenidos\_estacion.php”, “cont\_contenidos\_estacion\_video\_ver.php”, “cont\_contenidos\_estacion\_fotos\_ver.php”, “cont\_contenidos\_trayecto.php”, “cont\_contenidos\_trayecto\_video\_ver.php”, “cont\_contenidos\_trayecto\_fotos\_ver.php”.

En seleccionar l’enllaç de “contenidos” es carrega el fitxer “cont\_contenidos\_form.php” i es mostraran 2 panells desplegable, amb 2 formularis, un per estacions i un altre per trajectes, que permeten seleccionar una estació o un trajecte i els tipus de continguts que es volen en la cerca. Les funcionalitats son les mateixes tant si es tria una estació o un trajecte, però permet d’entrada fer un filtrat pel tipus de localització. A continuació s’explica el cas de escollir una estació per simplicitat donat que solament canviaran els fitxers que intervenen, i la taula de la base de dades a la que fan referència.

**Figura 31. Vista dels 2 panells per seleccionar estació i tipus de contingut.**

Un cop carregat el fitxer “cont\_contenidos\_form.php” cal seleccionar una estació del desplegable (select de formulari) on apareix el llistat de les estacions existents en l’aplicació, i al menys un dels tipus de continguts (vídeo, fotos o documents). Al seleccionar el botó “Buscar”, s’envia el formulari al fitxer “cont\_contenidos\_estacion.php”, i es farà una cerca dels tipus de continguts triats que fan referència a l’estació escollida, mostrant per pantalla tants desplegable com tipus escollits amb el resultat de la cerca.

**Contenidos por localización**

Estación de Cambrils:

Videos (1)

Selecciona un video y pulsa el boton :

|                                                          |                      |                         |                        |
|----------------------------------------------------------|----------------------|-------------------------|------------------------|
| Descripción:<br><input type="radio"/> Recorrido vagoneta | Fecha:<br>2013-12-12 | Especialidad:<br>varias | Elemento:<br>recorrido |
|----------------------------------------------------------|----------------------|-------------------------|------------------------|

Fotos (2)

Selecciona la foto o conjunto de fotos y pulsa el boton :

|                                                                 |                      |                               |                          |
|-----------------------------------------------------------------|----------------------|-------------------------------|--------------------------|
| <input type="checkbox"/> Descripción:<br>Señal 229              | Fecha:<br>2012-06-04 | Especialidad:<br>señalización | Elemento:<br>señal       |
| <input type="checkbox"/> Descripción:<br>Baliza previa señal E2 | Fecha:<br>2012-04-16 | Especialidad:<br>señalización | Elemento:<br>baliza ASFA |

**Figura 32. Detall del llistat amb el diferents continguts trobats.**

En fer clic al desplegable apareix un llistat de cadascun dels continguts trobats. En el cas de trobar algun vídeo, només caldrà seleccionar el input tipus radio del formulari i el botó “Ver” i s’enviarà el formulari al fitxer “cont\_contenidos\_estacion\_video\_ver.php” per poder-lo reproduir. En el cas de les fotos després de triar-ne una o més, cal seleccionar el botó “Ver”, i s’envia el formulari al fitxer “cont\_contenidos\_estacion\_fotos\_ver.php” on es mostrarà la galeria de les fotografies seleccionades. Per últim, des del llistat de documents, només cal seleccionar el botó “Ver” per obrir directament el document.

A diferència de la gestió de vídeos, fotos i documents, en el cas de seleccionar “contenidos”, no hi ha opcions de filtrat, encara que sempre es passa per el filtrat de saber si es tracta d’una estació o un trajecte, donat que la idea, és poder veure en un sol llistat, els continguts de qualsevol o de tots els tipus que hi ha en una localització concreta. Tampoc hi han les funcionalitats d’alta, modificació i baixa de continguts pels gestors i administradors.

#### 4.4.2. Implementació de la presentació de l’aplicació.

Per la presentació, es fa referència mitjançant l’ús de l’etiqueta <link> des del fitxer inicial de l’aplicació “index.php” al fitxer “css\_principal.css” que hi ha en la carpeta estils, juntament amb imatges com la que hi ha entre la capçalera i el menú, o les que serveixen pel fons dels laterals o de les zones del menú capçalera i peu. En aquest fitxer es defineixen els diferents estils, com per exemple els tipus de lletra, grandària, marges, alineació, o les llistes i el posicionament, i en general es declaren en aquest fitxer.

Quan es vol aplicar algun tipus, es fa referència al identificador o ID que identifica la etiqueta corresponent. Per citar algun exemple de codi, en el fitxer css\_principal.css s’ha declarat el identificador #contenidointerior, amb ,es següents propietats i valors:

```
#contenidointerior{
  background-color: #00E000;
  padding: 10px;
  margin: 10px;
  width: 938px;
  height: auto;
  text-decoration: none;
  font: 12px/100% verdana, arial, Helvetica, sans-serif;
  text-shadow: 0 1px 1px rgba(0,0,0,.3);
  -webkit-border-radius: .5em;
  -moz-border-radius: .5em;
  border-radius: .5em;
  -webkit-box-shadow: 0 1px 2px rgba(0,0,0,.2);
  -moz-box-shadow: 0 1px 2px rgba(0,0,0,.2);
  box-shadow: 0 1px 2px rgba(0,0,0,.2); }
```

Després des del codi HTML es fa referència a aquest identificador, per exemple amb:

```
<div id="contenidointerior" align="center">, i per tant al element se li aplicaran les propietats i valors definits .
```

Juntament amb el fitxer CSS, des del fitxer “index.php”, també es fa referència a les llibreries de jQuery i jQuery UI i als fitxers de JavaScript i de CSS de la galeria d’imatges GalleryView. Aquestes llibreries de codi JavaScript contenen processos o rutines que faciliten la tasca als programadors al estar implementades i provades, fent-les per exemple compatibles entre els diferents navegadors. A més de la galeria de fotos, la farem servir per modificar alguns estils o per mostrar o ocultar determinades parts en la part de continguts.

Com exemple de funcionament de les llibreries de jQuery, podem veure el següent codi que forma part d’un dels fitxers PHP. En ell es pot veure que quan es faci clic en els elements identificats amb #flip1 i #flip2 del selector jQuery #id, se li aplica el mètode slideToggle amb velocitat lenta als elements identificats amb #panel1 i #panel2. Això permetrà visualitzar o no (en funció de com estiguin) els diferents panells amb un efecte de desplaçament.

```
<script>
$(document).ready(function(){
  $("#flip1").click(function(){
    $("#panel1").slideToggle("slow"); });
  $("#flip2").click(function(){
    $("#panel2").slideToggle("slow"); });
});
</script>
```

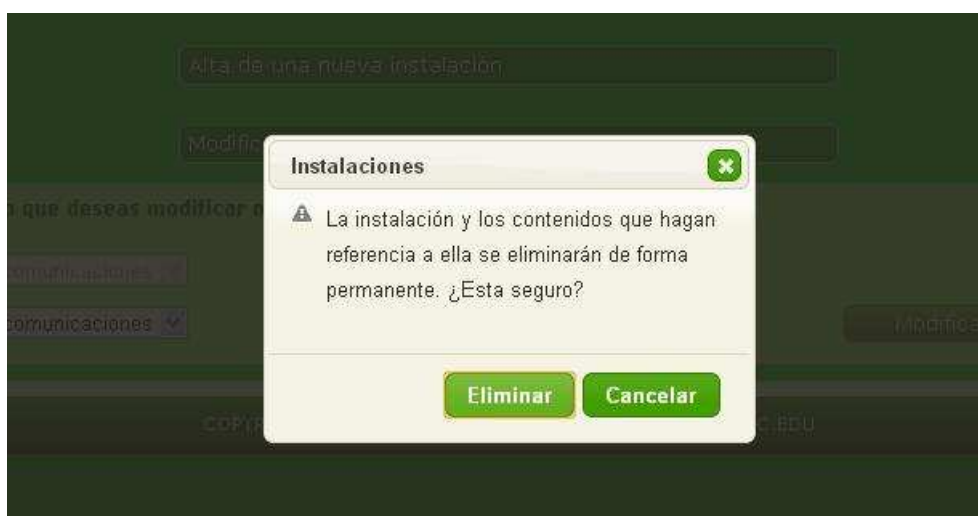
Les diferents funcionalitats es mostren a la part de continguts, en forma de formularis posicionats mitjançant capes contenidores amb etiquetes <div> i taules a les que se li apliquen els diferents estils com l’amplària, l’alineació o el color. Els formularis estan formats principalment per inputs de tipus “text”, “checkbox” i “radio” i de camps de

selecció “select”. També es fa servir el input tipus “password” en la pantalla de login, i el tipus “file” per pujar els fitxers al servidor.

Com exemple de codi podem veure el formulari de login que s’enviarà a una funció javascript per comprovar que els camps no estan buits abans d’enviar la informació:

```
<div id="login" >
  <h2 align="left">Login usuario</h2><hr>
  <form name="form_login" method="post"
action="apli/cont_login_valida_user.php">
    <table style="width:100%" align="center">
      <tr><td>Usuario:</td>
        <td><input type="text" id="username" name="username" required
/></td></tr>
      <tr><td>Password:</td>
        <td><input type="password" id="password" name="password" required/>
</td></tr>
      <tr><td align="center" colspan="2">
        <a href = "javascript:validar();" class="boto
green">Login</a></td></tr>
      <tr><td colspan="2" align="center"><div id="error"
style="color:red;"></div></td></tr>
    </table>
  </form>
</div>
```

Afegir per últim que quan es porten a terme les accions de gestió, tant dels diferents continguts, com de les instal·lacions, localitzacions i usuaris, es fan servir finestres modals per informar a l’usuari de les accions que porta a terme, i també en el cas de les accions d’eliminar, per confirmar que l’usuari esta segur de voler fer aquesta acció. Això s’ha implementat amb l’ús de jQuery UI, concretament amb el component Dialog:



**Figura 33. Vista de la finestra modal amb confirmació.**



Figura 34. Vista de la finestra modal amb un missatge.

Un altre component de jQuery UI utilitzat, ha estat DatePicker , per tal de mostrar un calendari i facilitar la introducció dels camps de data:



Figura 35. Vista del widget DatePicker de jQuery UI.

## 5. Conclusions i línees de futur

Una vegada implementada l'aplicació i realitzades les proves de funcionament, es comenten els resultats obtinguts per comparar-los amb els objectius a assolir per aquest projecte i s'exposen les conclusions personals de l'autor del projecte. A més, en aquest capítol també es comenten algunes de les possibles millores o línees de futur obertes.

### 5.1. Conclusions

Al inici d'aquest TFC es va definir l'objectiu troncal del projecte, consistent en el desenvolupament de una aplicació Web que permeti gestionar diferents continguts con vídeos, imatges i documents referents a la infraestructura ferroviària. Aquest objectiu ha estat assolit satisfactòriament, havent dissenyat una aplicació en la que totes les funcionalitats que es pretenien inicialment com son la gestió dels usuaris que tindran accés, la gestió de les instal·lacions ferroviàries i les localitzacions, o la gestió i visualització dels propis continguts han estat cobertes.

En general s'ha seguit la planificació proposada en la realització del projecte i s'han complert les fites previstes. Crec que ha estat molt important en la fase de disseny fer un bon plantejament de cadascuna de les funcionalitats i de la base de dades a implementar, donat que en principi tens la sensació de fer una feina que no serveix gaire, però que després comproves que simplifica molt les tasques de implementació.

També ha estat interessant la metodologia i la implementació de forma paral·lela de la lògica i de la presentació de l'aplicació

La realització del treball m'ha permès repassar coneixements adquirits durant la carrera (bases de dades i programació principalment) i aprofundir-ne amb l'aplicació de les noves tecnologies i del programari existent, mes actuals i recents per tal de poder desenvolupar a partir de coneixements bastant generals, una aplicació web amigable pels usuaris, intuïtiva i segura.

Una de les motivacions inicials al plantejar el projecte va ser obtenir un bon producte que pugues aprofitar en la meua feina actual (manteniment d'infraestructures ferroviàries), i considero que l'ús de l'aplicació aportarà beneficis interessants, al permetre centralitzar els continguts, evitant gran quantitat de còpies disperses que ocupen molt d'espai als usuaris i que moltes vegades quan es necessiten no es troben (ara sabran on buscar els continguts), poder veure la evolució temporal i comparatives de les instal·lacions, o inclús conèixer característiques de quins elements existien en la infraestructura ferroviària abans de produir-se un fet concret (com incidències, inundacions, accidents, etc.).

Comentar finalment que en les fases inicials de projecte i estat de l'art es va plantejar la possibilitat de implementar alguna funcionalitat que permetés fer un tractament dels vídeos emmagatzemats en l'aplicació, de manera que per exemple al reproduir un vídeo es parara en punts concrets per mostrar una determinada instal·lació relacionada, però tot i que ho vaig trobar factible, la dedicació al seu disseny i implementació hagués anat més enllà del temps disponible per a la realització del projecte. Crec que prendre la decisió en aquell moment de deixar aquesta possibilitat per a un possible treball futur va ser encertada, donat que les dues últimes setmanes en el disseny i implementació van ser molt intenses, i el primer pas i més important que volia assolir, era desenvolupar l'aplicació. En resum puc dir que estic satisfet de la feina feta.

## 5.2. Línees de futur

Els requeriments inicials pel l'aplicació de gestió de continguts multimèdia han estat complerts. Tot i això existeixen algunes millores o ampliacions que es podrien afegir i que podrien ser d'utilitat:

- Ampliació de les opcions de filtrat alhora de fer la cerca de continguts.
- Ampliar les opcions de veure vídeos amb formats no limitats per HTML5, per permetre en el cas de versions antigues de navegadors reproduir els vídeos en format flash per exemple.
- Implementar la possibilitat de afegir en l'aplicació el mateix vídeo en diferents formats per permetre la reproducció en més navegadors diferents.
- Implementar la possibilitat de donar de alta continguts en l'aplicació sense la necessitat de pujar els documents al servidor mitjançant un formulari, copiant prèviament els fitxers en un directori del servidor.
- Implementar una funcionalitat en la reproducció dels vídeos que permeti guardar diferents punts temporals, de forma que posteriorment es puguin seleccionar diferents opcions de reproducció o enllaçar amb altres continguts.

# Annexos

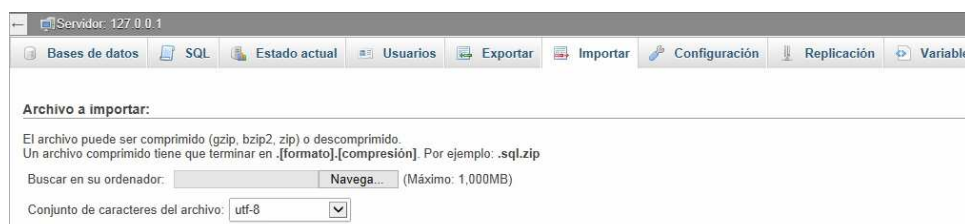
## Annex 1. Manual d'instal·lació i configuració de l'aplicació.

Per instal·lar l'aplicació cal seguir els següents passos:

Pas 1: Descarregar XAMPP, instal·lar-lo i configurar-lo, seguint els passos descrits en l'apartat 4.1.1

Pas 2: Els diferents continguts de les pàgines i fitxers han de estar col·locats sota el directori htdocs situat a C:\xampp\htdocs, si s'ha fet la instal·lació per defecte proposada. Llavors, cal descomprimir el fitxer ZIP de l'aplicació i copiar la carpeta “gestion”, sota aquest directori. Concretament la estructura d'arxius i carpetes un cop descomprimit serà C:\xampp\htdocs\gestion , i dins d'aquesta hi haurà l'arxiu “index.php” que servirà d'inici de l'aplicació i les carpetes “apli”, “cont”, “estils”, i “js”.

Pas 3: Des de la interfície de “phpMyAdmin” cal seleccionar la pestanya “Importar” i seleccionar l'arxiu “gestion.sql” que esta en el ZIP per crear la base de dades, les diferents taules i un usuari amb perfil d'administrador.



**Figura 36. Importació de la base de dades**

Pas 4: Des de la interfície anterior, seleccionant la pestanya “Usuarios” crear un nou usuari amb permisos (veure apartat 4.3), tenint en compte que les dades d'usuari i contrasenya han de coincidir amb les que apareixen al editar l'arxiu /gestion/apli/conexionBBDD.php.

Pas 5: Finalment, obrir el navegador web i introduir l'adreça corresponent a [http://IP\\_servidor/gestion](http://IP_servidor/gestion), o <http://localhost/gestion> des del propi ordinador on s'ha fet la instal·lació, i apareixerà la pantalla de login, introduir les dades de l'usuari i contrasenya del administrador per defecte (admin, admin) i ja es pot començar a utilitzar l'aplicació.



## Bibliografia

- [1] **Desenvolupament d'aplicacions Web:** Autor Carles Mateu, Fundació UOC. Material docent: Emmarcat dins del Màster internacional de programari lliure. 1a edició febrer 2006. Codi XP06/M2008/01165. (En línia publicat el 16-02-2010). [Última consulta:14-10-2013]. URL: <http://hdl.handle.net/10609/199> .
- [2] **HTML5: el nuevo estándar básico de la Web.** Autor: Jorge Franganillo. Anuario ThinkEPI, 2011, v. 5, pp. 261-265.
- [3] **HTML5:** Especificacions de HTML5 per part del W3C. (En línia). [Última consulta: 14-10-2013]. URL: <http://www.w3.org/TR/html5>.
- [4] **OpenLaszlo:** Web oficial del projecte OpenLaszlo. (En línia). [Última consulta: 14-10-2013].URL: <http://www.openlaszlo.org> .
- [5] **Adobe Flex:** Web oficial amb informació i enllaços de Adobe Flex, (En línia). [Última consulta: 14-10-2013]. URL: <http://www.adobe.com/es/products/flex.html> .
- [6] **HTML5:** Informació sobre el compliment de les especificacions de HTML5 per part dels navegadors i comparatives. (En línia). [Última consulta: 29-12-2013]. URL: <http://beta.html5test.com> .
- [7] **Apache:** Documentació oficial del servidor Web Apache. (En línia). [Última consulta: 23-11-2013]. URL: <http://httpd.apache.org> .
- [8] **MySQL:** Documentació oficial de MySQL. (En línia). [Última consulta: 10-12-2013]. URL: <http://www.mysql.com> .
- [9] **PHP:** Documentació oficial de PHP. (En línia). [Última consulta: 15-12-2013]. URL: <http://www.php.net> .
- [10] **Structured process you must know to develop a web application:** Desenvolupament d'aplicacions Web. Article publicat en el blog de Antonio Lupetti, any 2009. (En línia). [Última consulta:14-10-2013]. URL: <http://woork.blogspot.com.es/2009/01/structured-process-you-must-know-to.html> .
- [11] **Projecte Web:** Desenvolupament d'aplicacions Web. Autor: Alberto Otero García, Fundació UOC. Material docent: Emmarcat dins del Màster internacional de programari lliure. 1a edició febrer 2006. Codi 90.803\_a. (En línia publicat el 16-02-2010). [Última consulta:14-10-2013]. URL: <http://hdl.handle.net/10609/210> .
- [12] **XAMPP:** Web dels desenvolupadors del projecte, amb opcions de descarrega, documentació,etc. (En línia). [Última consulta: 28-11-2013]. URL: <http://www.apachefriends.org/es/xampp.html> .
- [13] **Gallery View:** Plugin jQuery per a la creació de galeries de fotos desenvolupat per Jack Wanders (En línia). [Última consulta:12-12-2013]. URL: <http://spaceforaname.com/galleryview>

- [14] **jQuery**: Llibreries de JavaScript jQuery amb la documentació, codi i exemples. (En línia). [Última consulta: 23-12-2013]. URL: <http://jquery.com> .
- [15] **jQuery UI**: Llibreries i components desenvolupats de Javascript jQuery user interface amb la documentació, codi i exemples. (En línia). [Última consulta: 23-12-2013]. URL: <http://jqueryui.com> .