

ESCUELA TÉCNICA SUPERIOR DE
INGENIERÍA DE TELECOMUNICACIÓN
UNIVERSIDAD POLITÉCNICA DE CARTAGENA



Trabajo Fin de Carrera

Desarrollo De Una Plataforma De Transmisión Y Generación De
Contenidos Audiovisuales



AUTOR: Alfonso Cánovas Baños
DIRECTOR: Juan Luis Predreño Molina
CO-DIRECTOR: Manuel Abrisqueta Martínez
Septiembre/2016

ESCUELA TÉCNICA SUPERIOR DE
INGENIERÍA DE TELECOMUNICACIÓN
UNIVERSIDAD POLITÉCNICA DE CARTAGENA

Titulación: Ingeniería de Telecomunicación

Reunido el tribunal examinador en el día de la fecha, constituido por:

D./D^a. _____

D./D^a. _____

D./D^a. _____

para juzgar el Proyecto Fin de Carrera titulado:

**Desarrollo De Una Plataforma De Transmisión Y Generación De
Contenidos Audiovisuales**

del alumno *D./D^a. Alfonso Cánovas Baños*
dirigido por *D./D^a. Juan Luis Predreño Molina*
co-dirigido por *D.D^a. Manuel Abrisqueta Martínez*

ACORDÓ POR _____ OTORGAR LA

CALIFICACIÓN DE _____

y, para que conste, se extiende firmada por los componentes del tribunal, la presente diligencia.

Cartagena, a ____ de ____ de ____

El Presidente:

El Vocal:

El Secretario:

Fdo.: _____ Fdo.: _____ Fdo.: _____

Universidad Politécnica de Cartagena
Escuela Técnica Superior de Ingeniería de
Telecomunicación

DESARROLLO DE UNA PLATAFORMA DE TRANSMISIÓN Y GENERACIÓN DE CONTENIDOS AUDIOVISUALES

REALIZADO POR

Alfonso Cánovas Baños

DIRIGIDO POR

Juan Luis Predreño Molina

CO-DIRIGIDO POR

Manuel Abrisqueta Martínez

Dpto. de: Tecnología de la Información y Comunicaciones (TIC)

Palabras clave: HTML5, CSS, PHP, jQUERY, MySQLI, CPCD, UPCT

Titulación: Ingeniería de Telecomunicación

Resumen:

El presente proyecto, realizado en el *centro de producción de contenidos digitales*, surge ante la necesidad de crear una plataforma web para la gestión de contenido multimedia.

Para la realización de esta tarea se ha optado por implementar, mediante diversos lenguajes de programación web, un gestor multimedia. Con la finalidad de solventar los requerimientos de organización en cuanto a contenidos audiovisuales del departamento.

Cartagena, 13 de septiembre de 2016

A todos los que me animaron en esta andadura,
a todos mis familiares y amigos.
Gracias por vuestro apoyo.

Alfonso Cánovas Baños

Acrónimos

ETSIT	Escuela Técnica Superior de Ingeniería de Telecomunicación
UPCT	Universidad Politécnica de Cartagena
HTML5	HyperText Markup Language, versión 5
PFC	Proyecto Fin de Carrera
PHP	Pre Hypertext Processor
jQuery	Bibliotecas de javascript
CSS	Cascading Style Sheets
AJAX	Asynchronous JavaScript And XML
MySQLI	Structured Query Language
CDN	Content Delivery Network

Índice

Acrónimos	VII
Prólogo	1
I Introducción	3
Introducción y visión general	5
Objetivo	6
Estado del arte	7
Metodología y directrices seguidas	7
Estructura del documento	8
Ámbito de aplicación	8
II Desarrollo del proyecto	9
1 Análisis de contenidos	11
1.1 Estructura y formato del proyecto	12
1.2 Introducción al estandar HTML5	13
1.3 Motivos para usar HTML5	14
1.4 Otros lenguajes empleados	15
1.4.1 PHP	15
1.4.2 CSS	15
1.4.3 Javascript	16
1.4.4 Bootstrap	16
1.5 Referencias bibliográficas	16
1.6 Formato de los Apéndices	16
2 Desarrollo de contenido	17
2.1 Software necesario	18
2.2 Instalación básica de XAMP	19

2.2.1	Instalar Apache	19
2.2.2	Instalar PHP	20
2.2.3	Base de datos bajo MySQL	21
2.3	Bootstrap	22
2.4	Diseño Responsive	24
2.4.1	JavaScript	25
3	Aplicación práctica	27
3.1	Introducción	28
3.2	Opciones de la plataforma multimedia	28
3.2.1	Registro / Acceso	28
3.2.2	Subir vídeo	30
3.2.3	Gestionar vídeos	31
3.2.4	Configurar perfil	32
3.2.5	Gestionar usuarios	33
3.2.6	Gestionar web	34
3.2.7	Salir	35
3.3	Categorías	35
3.4	Barra lateral	35
3.5	Pie de página	36
4	Desarrollo de contenido	37
4.1	Elementos del portal	38
4.1.1	CSS	38
4.1.2	Images	38
4.1.3	Includes	38
4.1.4	Javascript	39
4.1.5	php	39
4.2	Back-end	39
4.3	Front-end	40
4.4	HTACCESS	40
4.5	LESS CSS	42
4.6	Usos de CDN	43
4.7	Base de datos MySQLI	45
4.7.1	Base de datos para las categorías	45
4.7.2	Base de datos para los comentarios	46
4.7.3	Base de datos para la gestion web	46
4.7.4	Base de datos para la gestión de usuarios	47
4.7.5	Base de datos para la gestión de vídeos	48

III	Parte tercera.	49
	Conclusiones y líneas futuras	51
IV	Apéndices	53
A	Apéndice	55
A.1	Código fuente .htaccess	56
A.2	Código fuente variables.less	60
A.3	Código fuente header.inc	61
A.4	Código fuente managevideos.js	63
	Índice alfabético	66

Índice de figuras

1.1	Logo de HTML5	13
1.2	Logo PHP	15
2.1	Logo de Notepad++	18
2.2	Logo de XAMPP para Windows	19
2.3	Apache funcionando correctamente	20
2.4	PHP funcionando correctamente	21
2.5	Logo de Bootstrap	22
2.6	Diseño responsive	24
3.1	Modelo de usabilidad / accesibilidad	28
3.2	Modelo de opciones	29
3.3	Formulario de registro	29
3.4	Formulario de acceso	30
3.5	Formulario de subida de vídeo	30
3.6	Formulario de gestión de vídeo	31
3.7	Formulario de gestión de perfil	32
3.8	Formulario de gestión de usuarios	33
3.9	Configuración de tema	34
3.10	Configuración de categorías	34
3.11	Menú de categorías	35
3.12	Pie de página del portal	36
4.1	Estructura interna del portal	38
4.2	Comando habituales de .htaccess	41
4.3	Diagrama de categorías en la base de datos	45
4.4	Diagrama de comentarios en la base de datos	46
4.5	Diagrama de gestión web en la base de datos	46
4.6	Diagrama de gestión de usuarios en la base de datos	47
4.7	Diagrama de gestión de vídeos en la base de datos	48

Prólogo

Con la realización de este proyecto, se pretende dar a conocer a los alumnos, la manera mas rápida y eficiente de crear contenido web de tipo multimedia. Mediante un acercamiento a los principales sistemas de programación adaptados a los tiempos actuales.

A lo largo de este proyecto como resumen del trabajo realizado, podemos destacar las diferentes labores de instalación, implantación y administración del contenido multimedia y sus tecnologías allegadas, para conseguir las soluciones más óptimas según la recopilación de necesidades detectadas antes y durante el desarrollo del proyecto. Entre las tecnologías utilizadas resalta el aprendizaje y la utilización de **PHP**, **MySQL**, **HTML5** y **CSS**, que han resultado necesarios para el buen fin de este trabajo de fin de carrera.

De la misma forma y desde el presente documento, se anima a todo lector interesado en la programación y diseño web, que se lance a realizar cualquier nuevo proyecto, que no desista y sea paciente a la hora de trabajar. Todo esfuerzo lleva acompañado una satisfactoria recompensa.

Muchas gracias.

Parte I

Introducción

Introducción y visión general

Contenido

Objetivo	6
Estado del arte	7
Metodología y directrices seguidas	7
Estructura del documento	8
Ámbito de aplicación	8

Sinopsis

Hacer un diseño profesional, de forma totalmente autónoma y original. De esta forma podríamos enmarcar la idea de crear un nuevo contenido multimedia desde los cimientos, para la realización del presente documento.

El principal objetivo es dar a conocer la metodología, de manera que cualquiera pueda llegar a utilizarla de forma eficaz y en pocos pasos obteniendo conocimientos de forma rápida e intuitiva. ¹

Las metodologías y directrices seguidas se centran en qué procedimientos se han utilizado durante el desarrollo del proyecto. La forma del empleo y un acercamiento y conocimiento al desarrollo web.

¹En adelante, se utiliza la palabra *proyecto* como sinónimo de TFG/TFM/PFC, según se aplique (Nota del autor).

Objetivo

El principal objetivo de este proyecto final de carrera radica, en desarrollar una aplicación web como gestor de contenidos multimedia en el departamento de comunicación, la cual estará implementada mediante los lenguajes de programación web mas actuales.

La aplicación se llevará a cabo de forma que cubra las necesidades del departamento de comunicación en la **U.P.C.T.** Tratando de acercar al lector hacia una visión sencilla de desarrollo web para un contenido específico.

Entre las funcionalidades de la plataforma multimedia, podemos destacar como más importantes las enumeradas a continuación:

1. Crear una página donde el *departamento de comunicación* pueda subir los vídeos publicados y darles la máxima difusión posible.
2. Crear un acceso restringido, donde los miembros del departamento puedan gestionar sus publicaciones de forma segura.
3. Crear una sección de visionado general de las publicaciones donde estén a la vista desde el primer momento en el que el visitante accede a la web.

Para llevar a cabo los requerimientos descritos, necesitamos implementar un gestor de contenidos que nos permita editar y configurar según las necesidades especificadas. Después de sopesar varias opciones, se decide programar desde la base el citado gestor.

Así mismo, para el cumplimiento de estos objetivos, aparece un objetivo secundario evidente: realizar un aprendizaje sobre los lenguajes de programación que intervendrán en el diseño, para poder dar cabida a nuestros requisitos.

Además, como objetivo subyacente se buscará evaluar y demostrar el potencial que pueden ofrecer dichos lenguajes de programación, como tecnología para desarrollar páginas web de forma estructurada dentro del contexto del desarrollo tecnológico de la información en Internet.

Estado del arte

La aplicación que se desarrolla en este proyecto se encargará de gestionar los contenidos multimedia del departamento de comunicación y prensa de la U.P.C.T.

Actualmente no se dispone de ninguna página web activa para la gestión y publicación de contenido multimedia, donde poder comunicar la información considerada de interés en un mismo lugar, con los estándares de accesibilidad y de interoperatividad web actual como; suscripción, formularios de contacto, posibilidad de recibir feed-back de la comunidad a través de los comentarios, etc.

Debido al alto contenido multimedia que se genera en forma de video noticias, surge la necesidad de crear una plataforma web donde se pueda realizar la citada gestión y publicación de contenido.

Metodología y directrices seguidas

Para el *departamento de comunicación*, se crea una plataforma multimedia desde el estado más inicial, intentando cumplir con todas las necesidades que puedan surgir.

La forma de implementar esto a través de software, ha sido analizar los lenguajes de programación y herramientas web mas actuales y punteros, eligiendo el diseño que mejor se adapta al problema para conseguir los mejores resultados. Sumergirnos en estos lenguajes, nos dará una visión excepcionalmente valiosa acerca del *Estado del Arte* de esta rama de la tecnología de la información.

Estructura del documento

El documento está estructurado en varios capítulos que se detallan a continuación:

- **Introducción:** Este capítulo describe, a grandes rasgos, el problema al que se enfrenta el proyecto y la solución a implementar, además de definir la estructura de la memoria.
- **Conceptos previos y herramientas utilizadas:** En este capítulo se hace referencia a todo aquello que es necesario conocer para afrontar la realización del proyecto.
- **Desarrollo:** Este capítulo explica como está desarrollado el proyecto, describiendo la especificación técnica del mismo. Pretende ser una guía de como se ha realizado este trabajo de fin de carrera, para que sirva a la vez de testigo y punto de partida para cualquiera que necesite operar con la plataforma web del *departamento de comunicación*. Por ello, esta sección está desarrollada con gran cantidad de información gráfica que considero puede ayudar a utilizar este proyecto y modificarlo para futuras aplicaciones.
- **Síntesis del trabajo:** En este capítulo se resumen los objetivos alcanzados en este proyecto, dificultades encontradas, así como posibles mejoras y complementaciones al proyecto.
- **Bibliografía:** Apartado en el que se citan las fuentes consultadas durante el desarrollo del proyecto.

Ámbito de aplicación

Por último, completando los apartados anteriores, se explican las áreas de las que se compone el proyecto.

Parte II

Desarrollo del proyecto

Capítulo 1

Análisis de contenidos

Contenido

1.1	Estructura y formato del proyecto	12
1.2	Introducción al estandar HTML5	13
1.3	Motivos para usar HTML5	14
1.4	Otros lenguajes empleados	15
1.4.1	PHP	15
1.4.2	CSS	15
1.4.3	Javascript	16
1.4.4	Boostrap	16
1.5	Referencias bibliográficas	16
1.6	Formato de los Apéndices	16

Sinopsis

Se abarca el diseño del PFC, abordando los temas de mas interés para el usuario medio. Preguntas como ¿por qué se elige un determinado lenguaje? y ¿como se ha implementado el desarrollo? quedaran resueltas en el presente capitulo así como los aspectos principales y el asentamiento de las bases para comenzar a utilizar la plataforma, tras un rápido conocimiento de la misma.

1.1. Estructura y formato del proyecto

Éste capítulo describe las diferentes herramientas que se han utilizado para el desarrollo del proyecto y da una breve introducción a las tecnologías empleadas. Así mismo se explicaran los aspectos mas importantes para la instalación y el empleo de una herramienta de desarrollo web y sus opciones mas comunes.

En el documento resultante, el proyecto se estructura en los apartados siguientes:

- **Primeras páginas.** Contendrá las páginas que correspondan con el tipo de trabajo y que es específica de cada tipo de proyecto.
- **Agradecimientos y dedicatoria.** La sección de agradecimientos define un apartado donde el proyectando hace referencia a aquellas personas y/o instituciones que, de manera generosa, han colaborado de algún modo en la gestación y desarrollo del PFC. Igualmente el proyectando también hace mención a aquellos familiares, compañeros, amigos, etc., de los que ha recibido apoyo personal, moral o afectivo.
- **Índice.** Índice de figuras e índice de tablas, numerado automáticamente con su correspondiente página.
- **Capítulo de Introducción.** En éste capítulo se hará una introducción al ámbito del trabajo, comenzando por el contexto general y aproximándose progresivamente a los aspectos más concretos que se traten. Haciendo una exposición clara de los objetivos que se persiguen, la metodología empleada y el contenido del proyecto.
- **Capítulos de desarrollo.** Muestran el desarrollo de los distintos capítulos o epígrafes que componen el trabajo.
- **Conclusiones.** En este apartado se expondrá con claridad los resultados o conclusiones obtenidas y el juicio crítico que le merecen.
- **Referencias bibliográficas.** A continuación de las conclusiones se insertarán un apartado con todas las referencias bibliográficas que aparezcan en la memoria: revistas, artículos de revistas o congresos, páginas web, etc.
- **Apéndices.** Finalmente se presentarán los apéndices, si los hubiere.

1.2. Introducción al estandar HTML5

Para entender el desarrollo de la plataforma multimedia diseñada, se debe entender el lenguaje de programación base sobre el que se ha levantado la estructura que dará forma al proyecto. En este caso se trata de HTML5.

Se puede describir **HTML5** como un lenguaje usado para estructurar y presentar el contenido de cualquier pagina web. Se trata de uno de los estilos fundamentales para el funcionamiento de los sitios, pero no es el primero. En esta ocasión no encontramos ante la quinta revisión del estándar, el cual fue creado en 1990. Con HTML5, es posible hacer diseños usando menos recursos lo cual es un añadido para un desarrollo eficiente.

Se trata de un sistema para formatear el layout de nuestras páginas, así como hacer algunos ajustes a su aspecto. Con HTML5, los navegadores como **Firefox**, **Chrome**, **Explorer**, **Safari** y más pueden saber cómo mostrar una determinada página web, saber dónde están los elementos, dónde poner las imágenes, dónde ubicar el texto.



Figura 1.1: Logo de HTML5

El diseño mediante HTML5 es especialmente idóneo para construir y gestionar comunidades en Internet. No obstante, su flexibilidad y adaptabilidad, así como la gran cantidad de módulos adicionales disponibles, hace que sea adecuado para realizar muchos tipos diferentes de sitio web.

1.3. Motivos para usar HTML5

La idea detrás de HTML5 es que podamos visualizar el contenido multimedia variado que podemos encontrar en internet, aún cuando nos encontramos en dispositivos de gama baja que no podrían soportarlo cuando tienen que instalar infinidad de plug-ins. No solamente contamos con etiquetas especiales como audio, video y canvas, sino también integración con contenidos de gráficos en vectores. Con estas etiquetas, los usuarios pueden consumir videos y canciones, por ejemplo, sin necesidad de instalar nada de forma adicional.

Las más importantes de las nuevas etiquetas creadas son:

1. **Article**: esta etiqueta sirve para definir un artículo, un comentario de usuario o una publicación independiente dentro del sitio.
2. **Header, footer**: estas etiquetas individuales ahorran tener que insertar IDs para cada uno, como se solía hacer anteriormente. Además, se pueden insertar headers y footers para cada sección, en lugar de tener que hacerlo únicamente en general.
3. **Nav**: la navegación puede ser insertada directamente en el markup, entre estas etiquetas, que nos permitirán hacer que nuestras listas oficien de navegación.
4. **Section**: con esta etiqueta, una de las más importantes de las novedades, se puede definir todo tipo de secciones dentro de un documento. Por ponerlo de forma sencilla, funciona de una forma similar a la etiqueta div que nos separa también diferentes secciones.
5. **Audio y video**: estas son las dos más importantes etiquetas de HTML5, dado que nos permiten acceder de forma más simple a contenido multimedia que puede ser reproducido por casi todo tipo de dispositivos; marcan el tipo de contenido que estará en su interior.
6. **Embed**: con esta etiqueta se puede marcar la presencia de un contenido interactivo o aplicación externa.

1.4. Otros lenguajes empleados

A continuación se enumeran algunas de las herramientas utilizadas para la elaboración de la plataforma multimedia.

1.4.1. PHP

Es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Generalmente se ejecuta en un servidor web, tomando el código en PHP como su entrada y creando páginas web como salida. Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin coste.



Figura 1.2: Logo PHP

En este proyecto ha sido usado para la creación de la interface gráfica de la plataforma multimedia diseñada.

1.4.2. CSS

Es una tecnología que nos permite crear la estética de las páginas web de una manera más exacta, flexible y accesible que con tecnologías anteriores. CSS son las siglas de Cascading Style Sheets, en español Hojas de estilo en Cascada. Las hojas de estilo consisten en la creación de estilos propios dentro de las páginas web, o en un archivo separado que será invocado por el sistema al momento de presentar los documentos.

Gracias a las CSS podemos incluir márgenes, tipos de letra, fondos, colores... independientemente del contenido de la web pudiendo cambiar la apariencia de la misma sin necesidad de tocar para nada el contenido. Con esta tecnología quedan ambos separados completamente pudiendo realizar la labor de adaptación a distintos

dispositivos de forma completamente independiente al contenido mostrado.

En este proyecto ha sido usado para la creación de la interface gráfica de la Web que se ha realizado.

1.4.3. Javascript

JavaScript es un lenguaje de programación interpretado que se ejecuta en el navegador del cliente.

- **JQuery** Consiste en una librería de Javascript que simplifica las siguientes tareas comunes en el desarrollo web; manipulación de HTML/DOM, manipulación de CSS, eventos HTML, efectos y animaciones, AJAX.
- **Ajax** Usando la librería JQuery, hemos utilizado la técnica Asynchronous JavaScript And XML para realizar consultas al servidor de información de forma asíncrona.

1.4.4. Bootstrap

Bootstrap es un framework desarrollado por Twitter para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como, extensiones de JavaScript opcionales adicionales.

1.5. Referencias bibliográficas

A lo largo del proyecto, debe referenciarse el origen bibliográfico de todos los datos necesarios. Para eso, se almacenan fuentes bibliográficas (libros, artículos, revistas, páginas web, seminarios, etc.) en un archivo y formato estándar, para referenciarlas mediante etiquetas y variar el estilo bibliográfico según se prefiera.

1.6. Formato de los Apéndices

Se incluye en los apéndices información complementaria al texto del proyecto que no se considera indispensable para su comprensión. Los apéndices se numeran alfabéticamente y para su redacción, se han de seguir las reglas de seccionado como en los capítulos anteriores.

Capítulo 2

Desarrollo de contenido

Contenido

2.1	Software necesario	18
2.2	Instalación básica de XAMP	19
2.2.1	Instalar Apache	19
2.2.2	Instalar PHP	20
2.2.3	Base de datos bajo MySQL	21
2.3	Bootstrap	22
2.4	Diseño Responsive	24
2.4.1	JavaScript	25

Sinopsis

Se trataran los distintos lenguajes de programación dando un conocimiento de los mismos. De manera que se obtenga una visión general acerca del software empleado durante el desarrollo de la plataforma.

2.1. Software necesario

Para la realización de cada elemento de la plataforma multimedia, se ha optado por emplear **Notepad++**. Se trata de un editor de texto y código fuente libre con soporte para varios lenguajes de programación. Se parece al Bloc de notas en cuanto al hecho de que puede editar texto sin formato y de forma simple. No obstante, incluye opciones más avanzadas que pueden ser útiles para usuarios avanzados como desarrolladores y programadores. Se distribuye bajo los términos de la Licencia Pública General de GNU.



Figura 2.1: Logo de Notepad++

PHP, **MYSQL** y **APACHE** son conocidos habitualmente por las siglas **AMP** y existen para todas las plataformas de forma individual o bajo un paquete de fácil instalación que nos permitirán implementar estos tres elementos de forma sencilla en nuestro computador.

Si usamos Windows hemos de buscar en la web por **XAMPP**, en el caso de Linux dispondremos de un paquete para instalar en nuestros repositorios llamado **LAMP**, y por último, si usamos Macintosh nos será fácil bajar e instalar **MAMP**.

Durante el desarrollo de este documento se ha optado por emplear XAMPP bajo un entorno **Windows 10**.

2.2. Instalación básica de XAMPP

Tal y como se ha comentado en el apartado anterior, para nuestro desarrollo necesitamos instalar XAMPP



Figura 2.2: Logo de XAMPP para Windows

Para instalar XAMPP, bastará con descargar la aplicación de su página web y seguir los pasos de la instalación.

Una vez instalado accederemos al panel de control tal y como se muestra a continuación:

2.2.1. Instalar Apache

Ahora toca instalar el servidor web Apache. Bastará con ejecutar la acción de start el panel de control para la ejecución del servidor. Para corroborar el correcto funcionamiento se accederá desde el navegador a la dirección *"localhost"*.

Al acceder aparecerá el mensaje de Apache diciendo que está funcionando (It works!)

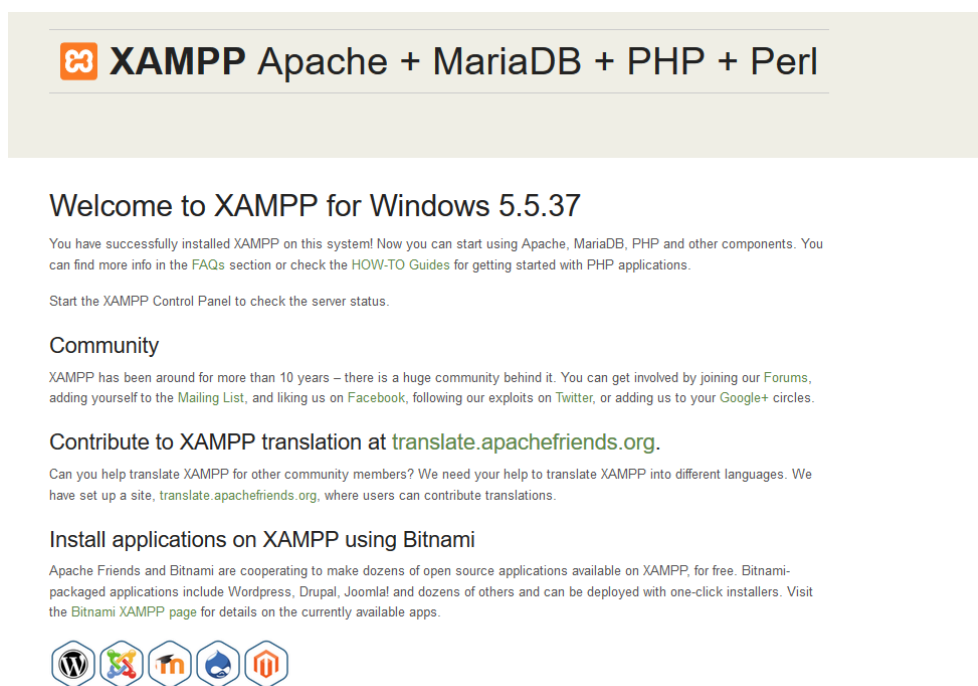


Figura 2.3: Apache funcionando correctamente

2.2.2. Instalar PHP

Gracias a XAMPP, PHP se integra en la instalación del mismo. PHP (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.

Lo que distingue a PHP de algo del lado del cliente como Javascript es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente. El cliente recibirá el resultado de ejecutar el script, aunque no se sabrá el código subyacente que era. El servidor web puede ser configurado incluso para que procese todos los ficheros HTML con PHP, por lo que no hay manera de que los usuarios puedan saber qué se tiene debajo de la manga.

Lo mejor de utilizar PHP es su extrema simplicidad para el principiante, pero a su vez ofrece muchas características avanzadas para los programadores profesionales. No sienta miedo de leer la larga lista de características de PHP. En unas pocas horas podrá empezar a escribir sus primeros scripts.

PHP Version 5.6.23-2+deb.sury.org~xenial+1	
System	Linux fonoal-VirtualBox 4.4.0-28-generic #47-Ubuntu SMP Fri Jun 24 10:09:13 UTC 2016 x86_64
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/5.6/apache2
Loaded Configuration File	/etc/php/5.6/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/5.6/apache2/conf.d
Additional .ini files parsed	/etc/php/5.6/apache2/conf.d/10-mysqld.ini, /etc/php/5.6/apache2/conf.d/10-opcache.ini, /etc/php/5.6/apache2/conf.d/10-pdo.ini, /etc/php/5.6/apache2/conf.d/15-xml.ini, /etc/php/5.6/apache2/conf.d/20-calendar.ini, /etc/php/5.6/apache2/conf.d/20-ctype.ini, /etc/php/5.6/apache2/conf.d/20-curl.ini, /etc/php/5.6/apache2/conf.d/20-dom.ini, /etc/php/5.6/apache2/conf.d/20-exif.ini, /etc/php/5.6/apache2/conf.d/20-fileinfo.ini, /etc/php/5.6/apache2/conf.d/20-ftp.ini, /etc/php/5.6/apache2/conf.d/20-gd.ini, /etc/php/5.6/apache2/conf.d/20-gettext.ini, /etc/php/5.6/apache2/conf.d/20-iconv.ini, /etc/php/5.6/apache2/conf.d/20-intl.ini, /etc/php/5.6/apache2/conf.d/20-json.ini, /etc/php/5.6/apache2/conf.d/20-mbstring.ini, /etc/php/5.6/apache2/conf.d/20-mcrypt.ini, /etc/php/5.6/apache2/conf.d/20-mysql.ini, /etc/php/5.6/apache2/conf.d/20-mysqli.ini, /etc/php/5.6/apache2/conf.d/20-pdo_mysql.ini, /etc/php/5.6/apache2/conf.d/20-phar.ini, /etc/php/5.6/apache2/conf.d/20-posix.ini, /etc/php/5.6/apache2/conf.d/20-readline.ini, /etc/php/5.6/apache2/conf.d/20-shmop.ini, /etc/php/5.6/apache2/conf.d/20-simplexml.ini, /etc/php/5.6/apache2/conf.d/20-sockets.ini, /etc/php/5.6/apache2/conf.d/20-sysmsg.ini, /etc/php/5.6/apache2/conf.d/20-sysvsem.ini, /etc/php/5.6/apache2/conf.d/20-sysvshm.ini, /etc/php/5.6/apache2/conf.d/20-tokenizer.ini, /etc/php/5.6/apache2/conf.d/20-wddx.ini, /etc/php/5.6/apache2/conf.d/20-xmldr.ini, /etc/php/5.6/apache2/conf.d/20-xmlwriter.ini, /etc/php/5.6/apache2/conf.d/20-xsl.ini

Figura 2.4: PHP funcionando correctamente

2.2.3. Base de datos bajo MySQL

Para la implementación de la base de datos, es necesario utilizar una base de datos que recoja los formularios de registro de los usuarios. En nuestro caso al emplear XAMPP contamos de base con la integración de MySQL para nuestro propósito.

MySQL es un sistema gestor de bases de datos (SGBD, DBMS por sus siglas en inglés) muy conocido y ampliamente usado por su simplicidad y notable rendimiento. Aunque carece de algunas características avanzadas disponibles en otros SGBD del mercado, es una opción atractiva tanto para aplicaciones comerciales, como de entretenimiento precisamente por su facilidad de uso y tiempo reducido de puesta en marcha. Esto y su libre distribución en Internet bajo licencia GPL le otorgan como beneficios adicionales (no menos importantes) contar con un alto grado de estabilidad y un rápido desarrollo.

Ahora iniciamos una conexión http al servidor, ingresando localhost/phpmyadmin en la barra de direcciones, y veremos que se nos ofrece la posibilidad de seleccionar la configuración de mysql deseada para nuestro entorno.

2.3. Bootstrap

Bootstrap, es un framework originalmente creado por **Twitter**, que permite crear interfaces web con **CSS** y **JavaScript**, cuya particularidad es la de adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice. Es decir, el sitio web se adapta automáticamente al tamaño de un PC, una Tablet u otro dispositivo.

Esta técnica de diseño y desarrollo se conoce como **responsive design** o diseño adaptativo. Esta será la base de la plataforma multimedia a desarrollar pues contendrá un diseño totalmente responsive.



Figura 2.5: Logo de Bootstrap

El beneficio de usar responsive design en un sitio web, es principalmente que el sitio web se adapta automáticamente al dispositivo desde donde se acceda. Uno de los elementos mas usado, es el empleo de media queries, el cual se trata de un módulo **CSS3** que permite la representación de contenido para adaptarse a condiciones como la resolución de la pantalla de esta forma, puedes tener una web muy fluida capaz de adaptarse a casi cualquier tamaño de forma automática.

Aun ofreciendo todas las posibilidades que ofrece Bootstrap a la hora de crear interfaces web, los diseños creados con Bootstrap son simples, limpios e intuitivos, esto les da agilidad a la hora de cargar y al adaptarse a otros dispositivos. El framework trae varios elementos con estilos predefinidos fáciles de configurar: *botones, menús desplegables, formularios incluyendo todos sus elementos e integración jQuery* para ofrecer ventanas y tooltips dinámicos.

Bootstrap tiene un soporte relativamente incompleto para HTML5 y CSS 3, pero es compatible con la mayoría de los navegadores web. La información básica de compatibilidad de sitios web o aplicaciones esta disponible para todos los dispositivos y navegadores. Existe un concepto de compatibilidad parcial que hace disponible la información básica de un sitio web para todos los dispositivos y navegadores.

Por ejemplo, las propiedades introducidas en CSS3 para las esquinas redondeadas, gradientes y sombras son usadas por Bootstrap a pesar de la falta de soporte de navegadores antiguos. Esto extiende la funcionalidad de la herramienta, pero no es requerida para su uso.

Desde la versión 2.0 también soporta diseños sensibles. Esto significa que el diseño gráfico de la página se ajusta dinámicamente, tomando en cuenta las características del dispositivo usado (Computadoras, tabletas, teléfonos móviles).

Existen diferentes formas de empezar a utilizar Bootstrap, dependiendo de la necesidad del usuario, se tienen:

1. **Descargar el código CSS y JavaScript compilado**, que es la forma más sencilla de empezar a utilizar Bootstrap. La desventaja es que esta versión no incluye ni los archivos originales ni la documentación. Para descargar esta versión, accede a getbootstrap.com y pulsa el botón Download Bootstrap.
2. **Descargar el código fuente**, contiene todos los archivos Less, y JavaScript originales de Bootstrap. La desventaja es que requiere un compilador de archivos Less y cierto trabajo de configuración. Descarga la versión más reciente en el sitio github.com/twbs/bootstrap/releases.
3. **Descargar el código fuente en formato Sass**, se trata de una variante de la versión anterior y que se ha creado para facilitar la integración de Bootstrap en las aplicaciones Ruby On Rails, Compass o cualquier otro proyecto basado en Sass. Descarga la versión más reciente en el sitio github.com/twbs/bootstrap-sass/releases.

Durante el desarrollo de la plataforma se realizaron diversas llamadas a código Bootstrap compilado por lo cual era conveniente su descripción en este apartado. Así como remarcar que se realizará un diseño totalmente responsive.

2.4. Diseño Responsive

El diseño web adaptable es una configuración en la que el servidor siempre envía el mismo código HTML a todos los dispositivos y se utiliza código CSS para modificar el procesamiento de la página en el dispositivo. Para indicar a los navegadores que



Figura 2.6: Diseño responsive

la página se adaptará a todos los dispositivos, se añade una meta-etiqueta en el encabezamiento del documento:

Listing 2.1: Metaetiqueta

```
<meta name="viewport" content="width=device-width,
initial-scale=1.0">
```

La metaetiqueta viewport da instrucciones al navegador sobre cómo ajustar las dimensiones y la escala de la página a la anchura del dispositivo. Cuando el metaetiqueta viewport no está presente, los navegadores para móviles automáticamente procesan la página con una anchura de pantalla propia de ordenadores (por lo general, unos 980 px, aunque puede variar según el dispositivo).

A continuación, los navegadores para móviles intentan mejorar el aspecto del contenido aumentando los tamaños de fuente, y adaptando el tamaño del contenido para que se ajuste a la pantalla o mostrando solo la parte del contenido que cabe en pantalla.

Se recomienda el uso del diseño web adaptable porque ofrece los siguientes aspectos positivos:

- Si se usa una sola URL, a los usuarios les resulta más fácil compartir el contenido y enlazarlo.
- Ayuda a los algoritmos de Google a asignar correctamente las propiedades de indexación a la página en vez de tener que señalar la existencia de páginas correspondientes para ordenadores o para móviles.
- Requiere menos tiempo de ingeniería para mantener varias páginas para el mismo contenido.
- Reduce la posibilidad de que se produzcan los errores habituales que afectan a los sitios para móviles.
- No es necesario realizar un redireccionamiento para que los usuarios lleguen a la vista optimizada para su dispositivo, de modo que se reduce el tiempo de carga. Además, el redireccionamiento basado en el agente de usuario tiende a causar errores y puede empeorar la experiencia del usuario en tu sitio (consulta la sección Dificultades al detectar agentes de usuario para obtener más detalles).
- Ahorra recursos cuando Googlebot rastrea tu sitio. En el caso de las páginas con diseño web adaptable, cualquier agente de usuario de Googlebot solo tiene que rastrear tu página una vez para recuperar todas las versiones del contenido (en vez de tener que rastrearlas varias veces con diferentes agentes de usuario de Googlebot). Esta mejora en la eficiencia de rastreo puede ayudar de forma indirecta a Google a indexar una parte mayor del contenido del sitio y a mantenerlo adecuadamente actualizado.

2.4.1. JavaScript

Un tema que se debería tener en cuenta a la hora de crear sitios optimizados para móviles es el uso de JavaScript para modificar el modo en el que se procesa y se comporta el sitio en distintos dispositivos. JavaScript suele utilizarse, por ejemplo, para determinar qué anuncio o qué resolución alternativa de imagen se mostrará en la página.

En esta sección se describen las diferentes formas de usar JavaScript y cómo se ajustan al diseño web adaptable que Google recomienda.

A continuación se indican tres configuraciones populares de JavaScript en sitios optimizados para móviles:

- JavaScript adaptable: en esta configuración, todos los dispositivos reciben el mismo contenido HTML, CSS y JavaScript. Cuando el código JavaScript se ejecuta en el dispositivo, el procesamiento o el comportamiento de este sitio cambia. Si un sitio web necesita JavaScript, esta es la configuración recomendada por Google.
- Detección combinada: en esta configuración, el sitio utiliza JavaScript y la detección de servidor de la capacidad del dispositivo para mostrar contenido distinto en función del dispositivo.
- JavaScript dinámico: en esta configuración, todos los dispositivos reciben el mismo código HTML, pero el código JavaScript se envía desde una URL que ofrece de forma dinámica un código JavaScript adaptado al agente de usuario del dispositivo.

Capítulo 3

Aplicación práctica

Contenido

3.1	Introducción	28
3.2	Opciones de la plataforma multimedia	28
3.2.1	Registro / Acceso	28
3.2.2	Subir vídeo	30
3.2.3	Gestionar vídeos	31
3.2.4	Configurar perfil	32
3.2.5	Gestionar usuarios	33
3.2.6	Gestionar web	34
3.2.7	Salir	35
3.3	Categorías	35
3.4	Barra lateral	35
3.5	Pie de página	36

Sinopsis

Se tratarán los distintos lenguajes de programación dando un conocimiento de los mismos. De manera que se obtenga una visión general acerca del software empleado durante el desarrollo de la plataforma.

3.1. Introducción

La metodología es un conjunto de técnicas que nos proporcionan la manera de proceder para conseguir unos propósitos deseados en un producto. En este proyecto se ha utilizado la metodología adecuada para desarrollar un sistema interactivo bajo los parámetros de la usabilidad y la accesibilidad, ya que el fin último del proyecto es que sea útil para un determinado grupo de personas que necesitan ciertos requisitos de accesibilidad.

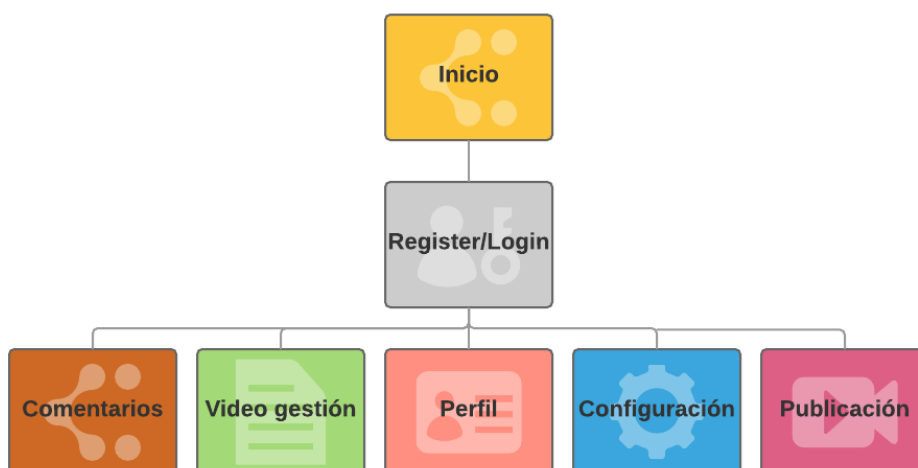


Figura 3.1: Modelo de usabilidad / accesibilidad

3.2. Opciones de la plataforma multimedia

A continuación se detallan las distintas opciones implementadas en el desarrollo de la plataforma las cuales estarán disponibles para los usuarios de la misma.

3.2.1. Registro / Acceso

Se trata del formulario de acceso a la plataforma multimedia mediante logueo previo registro. Se deberán rellenar los formularios solicitados para acceder a la aplicación.

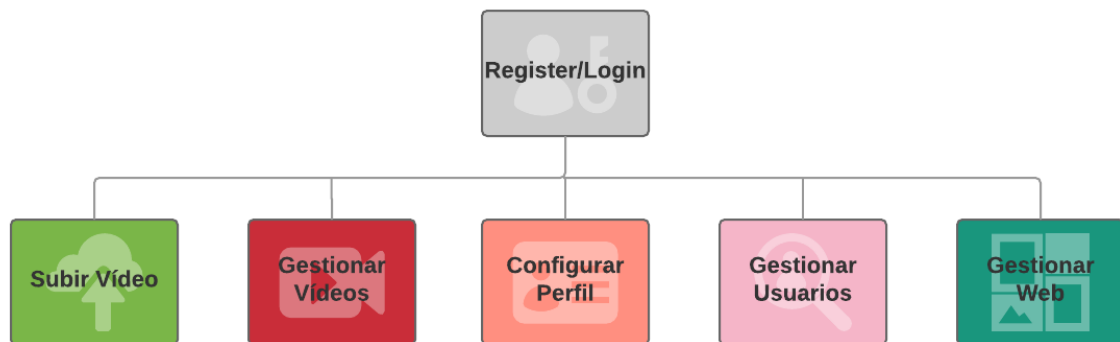


Figura 3.2: Modelo de opciones

Regístrate con nosotros [Acceso](#) [Registro](#)

Nombre de usuario

Correo electrónico

Contraseña

Comprobar contraseña

Estoy de acuerdo con los terminos de uso

[Cerrar](#) [Registrarse](#)

Figura 3.3: Formulario de registro

Inicia sesión con nosotros Acceso Registro

Nombre de usuario o correo
Inserta tu nombre de usuario o correo

Contraseña
Inserta tu contraseña

Cerrar Recordar contraseña Acceder

Figura 3.4: Formulario de acceso

3.2.2. Subir vídeo

En esta opción tenemos se realizará el formulario para la subida de un determinado vídeo a la plataforma multimedia, rellenando una serie de campos:

Sube un vídeo

A Título del vídeo:
Inserta un título aquí

Imagen del vídeo:
Inserta una imagen aquí

Link del vídeo:
Inserta un link aquí

Descripción del vídeo:
Inserta una descripción aquí

Categoría del vídeo:
Selecciona una categoría

Estoy de acuerdo con los terminos y condiciones de uso mostradas [aquí](#).


Subir vídeo

Figura 3.5: Formulario de subida de vídeo

- **Título del vídeo:** Se introducirá el sobrenombre del vídeo que se desea publicar en el portal multimedia.
- **Imagen del vídeo:** Se colocará el link de la imagen donde se encuentre hospedada; esta imagen debera tener un tamaño de 450x353px.
- **Link del vídeo:** Dirección que hace referencia al vídeo que se desea publicar.
- **Descripción del vídeo:** Breve descripción del vídeo que se desea publicar.
- **Categoría del vídeo:** Categoría del vídeo donde se va a publicar.

3.2.3. Gestionar vídeos

En este apartado se podrán gestionar los vídeos publicados por los usuarios, según el nivel del mismo. En tal caso el administrador o moderador de la plataforma podrán publicar lo vídeos subidos y gestionarlos. El apartado cuenta con los siguientes elementos:

 **Gestionar vídeos**

Show entries Search:

Id	Título	Autor	Categoría	Comentarios	Fecha de creación	Gestionar
13	Alumnos de Caminos de la UPCT reproducen el comportamiento estructur...		Ciencia	0	5 de Septiembre	Gestionar
12	Estudiantes de Agrónomos de la UPCT plantan especies autóctonas en...		Ciencia	0	5 de Septiembre	Gestionar
11	Tecnología y ocio en las visitas a la UPCT del Campus Científico d...		Cine	0	5 de Septiembre	Gestionar
10	Cartagena, de ciudad militar a ciudad universitaria		Cine	0	5 de Septiembre	Gestionar
9	Cartagena Ciudad Universitaria		Ciencia	0	2 de Septiembre	Gestionar

Figura 3.6: Formulario de gestión de vídeo

- **Identificador (Id):** Se trata del identificador numérico del vídeo publicado.
- **Título:** Se trata del título del vídeo publicado.
- **Autor:** Se trata del nombre de usuario que ha publicado el vídeo.
- **Categoría:** Identifica la categoria en la que se ha publicado el vídeo.
- **Comentarios:** Identifica el numero total de comentarios que los usuarios de la plataforma hayan realizado sobre el vídeo publicado.

- **Fecha de creación:** Indica el día y mes en el que se ha publicado un determinado vídeo.
- **Gestionar:** Permite acceder a los datos del vídeo publicado para realizar gestiones sobre el mismo.

3.2.4. Configurar perfil

En esta sección el usuario podrá editar y configurar sus datos de usuario con los campos que se describen a continuación:

The image shows a web form titled "Perfil" with a user icon. It contains several input fields and buttons. The first section includes fields for "Nombre completo:" (with placeholder "Introduce tu nombre"), "Fecha de nacimiento:" (with value "2016-09-02 09:09:48"), "Imagen de perfil:" (with placeholder "Introduce el enlace a tu imagen de perfil"), and "Correo electrónico:". Below these is a large "Descripción:" text area. A button "Actualizar perfil" is located below the description. The second section includes fields for "Imagen de perfil:" and "Correo electrónico:", with a button "Actualizar contraseña" below them.

Figura 3.7: Formulario de gestión de perfil

- **Nombre completo:** Nombre del usuario
- **Fecha de nacimiento:** Indica la fecha de nacimiento del usuario.
- **Imagen de perfil:** Permite incorporar una imagen al perfil del usuario.
- **Correo electrónico:** Se trata de la dirección de correo del usuario.
- **Descripción:** Permite al usuario establecer una breve descripción de su persona.
- **Actualizar contraseña:** Facilita al usuario realizar el cambio de contraseña asociado a su cuenta.

3.2.5. Gestionar usuarios

Este apartado permite al administrador configurar los niveles de jerarquía de los usuarios de tal forma según el nivel de usuario; se podrán acceder a determinados apartados de gestión del portal multimedia:



Gestionar Usuarios

Id	Nick	Mail	Comentarios	Videos	Nivel
1	Ped		0	0	Admin
6	Alfonso		0	0	Admin
7			0	0	Admin

Nick de usuario:

[✖ Eliminar usuario](#) [⬇ Degradar a usuario](#) [👤 Nombrar moderador](#) [👑 Subir a admin](#)

Figura 3.8: Formulario de gestión de usuarios

- **Identificador (Id):** Se trata del identificador numérico del usuario.
- **Nick:** Se trata del nombre del usuario.
- **Mail:** Identifica la dirección de correo del usuario.
- **Comentarios:** Indica el número de comentarios publicados por el usuario.
- **Comentarios:** Enumera el número total de vídeos publicados por el usuario.
- **Nivel:** Muestra el nivel de operabilidad del usuario distinguiendo entre tres posibles niveles.

Administrador: Tendrá el control total de la plataforma.

Moderador: Podrá gestionar la publicación y comentarios de los usuarios.

Redactor: Podrá subir vídeos y comentarios a la plataforma previa moderación

3.2.6. Gestionar web

Se trata de una de las opciones más importantes de la plataforma multimedia, donde se podrán realizar diversas configuraciones que se detallan a continuación:

- **Configurar tema:** Se podrán actualizar diversos contenidos de la plataforma como redes sociales, políticas de cookies, avisos legales y descripción del contenido.

Configurar tema

Política de cookies:

Condiciones de uso:

Facebook:

Twitter:

Google Plus:

Youtube:

Vimeo:

Descripción:

El Centro de Producción de Contenidos Digitales (CPCD) de la Universidad Politécnica de Cartagena (UPCT), cuenta en la actualidad con el equipamiento necesario para la creación y experimentación en contenidos digitales tales como el tratamiento digital de imágenes, la producción y postproducción audiovisual, el análisis y tratamiento 3D, la animación 2D, la integración de medios, los servicios polimedia, las emisiones en streaming y la formación on-line a distancia.

Figura 3.9: Configuración de tema

- **Configura menú:** Se podrán gestionar las categorías presentes en la plataforma, de forma que podrán crearse nuevas categorías así como borrarse en caso de que se desee.

Configurar Menu

Lista de categorías:

Opciones:

Nueva categoría:

Opciones:

Figura 3.10: Configuración de categorías

3.2.7. Salir

Permite al usuario abandonar la sesión activa de su cuenta.

3.3. Categorías

Tanto en el menú contextual como en el formulario de publicación de nuevo contenido, es posible seleccionar la categoría del vídeo que deseamos visualizar, distinguiendo de tal forma varias secciones que se detallan a continuación:

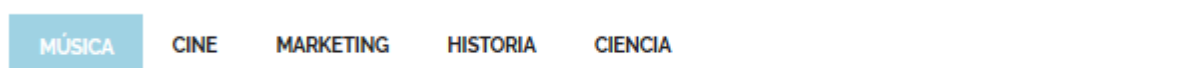


Figura 3.11: Menú de categorías

- Música: Contenido relacionado con entornos musicales.
- Cine: Contenido con todo lo relacionado con el mundo cinematográfico.
- Marketing: Contenido acerca de marketing.
- Historia: Contenido que muestra elementos de la historia.
- Ciencia: Contenidos de publicaciones relacionadas con el entorno de la ciencia.

3.4. Barra lateral

En este apartado se realiza una breve descripción en lo que a los elementos de la barra lateral se refiere dentro del portal multimedia:

- Únete a nosotros: Permite a los usuarios i/o visitantes seguir la publicación de contenido a través de las redes sociales; Facebook, Twitter, Google+, Youtube, Vimeo.
- Imagen de la universidad: Se muestra una imagen relevante de la universidad a efectos decorativos.
- Top vídeos: Se muestran enlaces a los vídeos mas visitados por los usuarios.

3.5. Pie de página

Elemento del portal dedicado a referencias y políticas de uso de la misma por parte de los usuarios tanto visitantes como registrados en el portal:

- Sobre nosotros: Breve descripción del marco en el que se encuadra el portal y descripción de sus creadores.
- Enlaces de interés: Diversos enlaces referenciados al entorno de la UPCT.
- Últimas noticias: Se muestran los títulos de los últimos vídeos publicados por los usuarios de la plataforma.
- Año: Se muestra el año actual de uso de la plataforma
- Política de cookies: Se detallan las políticas del uso de cookies a los visitantes del portal multimedia.
- Condiciones de uso: Se detallan las condiciones de uso del portal.



Figura 3.12: Pie de página del portal

Capítulo 4

Desarrollo de contenido

Contenido

4.1	Elementos del portal	38
4.1.1	CSS	38
4.1.2	Images	38
4.1.3	Includes	38
4.1.4	Javascript	39
4.1.5	php	39
4.2	Back-end	39
4.3	Front-end	40
4.4	HTACCESS	40
4.5	LESS CSS	42
4.6	Usos de CDN	43
4.7	Base de datos MySQLI	45
4.7.1	Base de datos para las categorías	45
4.7.2	Base de datos para los comentarios	46
4.7.3	Base de datos para la gestión web	46
4.7.4	Base de datos para la gestión de usuarios	47
4.7.5	Base de datos para la gestión de vídeos	48

Sinopsis

Se realiza una aproximación y explicación a los distintos archivos que forman la base para el funcionamiento del portal multimedia.

4.1. Elementos del portal

Se incluyen todos los elementos y archivos que dan vida al portal y forman la estructura y diseño del mismo:

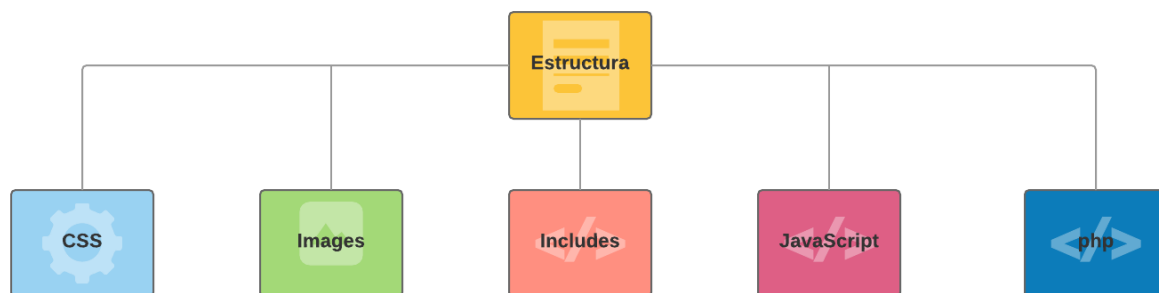


Figura 4.1: Estructura interna del portal

4.1.1. CSS

Esta carpeta tendrá todos tus códigos CSS usados para darle estilo al contenido del portal (por ejemplo, establecer el color de tus textos y de fondos).

4.1.2. Images

Esta carpeta almacenara todas las imágenes que se emplearan en el portal.

4.1.3. Includes

Cuando se incluye un archivo, el código que contiene hereda el ámbito de las variables de la línea en la cual ocurre la inclusión. Cualquier variable disponible en esa línea del archivo que hace el llamado, estará disponible en el archivo llamado, desde ese punto en adelante. Sin embargo, todas las funciones y clases definidas en el archivo incluido tienen el ámbito global. Se emplean los archivos `inc` para optimizar el coste computacional del portal.

4.1.4. Javascript

Esta carpeta tendrá todos los códigos JavaScript usados para darle interactividad y funcionalidad a tu sitio (por ejemplo. botones que cargan datos cuando hacen clic).

4.1.5. php

Esta carpeta tendrá todos los códigos php usados como por ejemplo para las llamadas a la base de datos.

4.2. Back-end

El **Back-End** es el área que se dedica a la parte lógica de nuestro portal web, es el encargado de que todo funcione como debería, es la parte de atrás que de alguna manera no es visible para el usuario ya que no se trata de diseño, o elementos gráficos, se trata de programar las funciones que tendrá nuestra plataforma multimedia. Se trata por tanto de la programación pura y dura, desde la programación de las funciones del sitio hasta bases de datos y otras opciones.

El Back-end trabaja todo el tiempo con lenguajes de programación, lenguajes que requieren de una lógica ya que esta área es también la encargada de optimizar recursos, de la seguridad de un sitio y demás. Cosas que el usuario no ve de primeras pero que existe código detrás que esta haciendo su trabajo.

Los lenguajes de programación que se utilizan en el Back-end en la actualidad son **PHP**, **Javascript**, **Phyton** y **Ruby**. Además de **HTML** y **CSS** simplemente para poder crear una estructura básica en la que se va a trabajar.

Las herramientas que se utilizan en el Back-end son editores de código, compiladores, algunos debuggeadores para revisar errores y seguridad, gestores de bases de datos y algunas otras cosas.

El flujo de trabajo de un back-end consiste en darle funciones a un sitio, por lo general el front-end hace un sitio estático, y el back-end después le da funciones y adapta el sistema programado a ese sitio web.

4.3. Front-end

Front-end es la parte del desarrollo web que se dedica de la parte frontal de un sitio web, en pocas palabras del diseño de un sitio web, desde la estructura del sitio hasta los estilos como colores, fondos, tamaños hasta llegar a las animaciones y efectos.

Se dedica básicamente al diseño web, pero esto no significa que no toque código, tanto el front-end como el back-end están en contacto con código todo el tiempo, dentro del área de front-end se trabaja con lenguajes mayormente del lado del cliente, como **HTML** y **CSS** para darle estructura y estilo al sitio, y **Javascript** para complementar los anteriores y darle dinamismo a los sitios web.

La parte del front-end es mas artística y creativa, una gran parte se centra en hacer sitios bonitos y amigables, resumamos en elementos visuales. El flujo de trabajo de front-end puede variar pero la mayoría del tiempo es diseñar bocetos, transformarlos en prototipos y por ultimo transformar la idea en un sitio web.

4.4. HTACCESS

El software de servidor **Apache**, sobre el que funciona la mayoría de la Web, ofrece configuración de directorios a través de los archivos de Acceso a Hipertexto, Hypertext Access en inglés, más conocidos como archivos .htaccess

Estos archivos **.htaccess** permiten ajustes personalizados y específicos para cada sitio de directivas de configuración del sistema, definidas en el archivo de configuración principal de Apache (httpd.conf).

Las reglas o directivas de .htaccess afectan al directorio superior y todos los sub-directorios donde esté situado. Así que para que se apliquen sus configuraciones a toda una web se coloca el archivo .htaccess en la carpeta raíz de sitio.

- **Errores:** Al ser un archivo de configuración, .htaccess es realmente potente, y en consecuencia peligroso si no se usa bien. Hasta el más ligero error de sintaxis, como un espacio de sobra, puede provocar un funcionamiento desastroso en el servidor.
- **Rendimiento:** Las directivas de .htaccess ofrecen configuraciones de directorio sin necesidad de acceder al archivo principal de configuración de

Apache (`httpd.conf`). Sin embargo, por cuestiones de rendimiento y seguridad, siempre que sea posible, se debe utilizar el archivo principal de configuración para las directivas del servidor.

Por ejemplo, cuando un servidor está configurado para procesar directivas de `.htaccess`, Apache debe buscar en cada directorio del dominio y cargar todos y cada uno de los archivos `.htaccess` en cada petición de documentos. Esto provoca un incremento en el tiempo de procesamiento de la página y, en consecuencia, perjudica el rendimiento.

- **Comandos habituales:**

^	– Indica el principio de la palabra a buscar.
\$	– Indica el final de la palabra a buscar.
.	– El punto, serviría para indicar cualquier carácter menos el espacio (ejemplo: a, B, 3).
*	– El asterisco indica, que el carácter previo, puede aparecer 0 o más veces.
()	– Los paréntesis permiten agrupar y guardar una parte de la primera cadena, para su uso posterior, utilizando la variable <code>\$1</code> en la segunda cadena. Esto es útil para manejar peticiones a ficheros en particular que deberían ser lo mismo tanto en la antigua como la nueva versión de la URL

Figura 4.2: Comando habituales de `.htaccess`

A continuación se explica brevemente el uso de este archivo en nuestra plataforma multimedia:

La principal idea de emplear este tipo de archivos es crear una url **amigable**. Se puede ilustrar con un ejemplo:

- URL no amigable: **`www.dominio.es/prueba.php?id=5`**
- URL amigable: **`www.dominio.es/prueba/video`**.

Las URL amigables son más fáciles de usar y de recordar por los usuarios y también se dice que son mejores para el posicionamiento en buscadores. Además describen mucho mejor los enlaces dentro de una página web y añade un

mayor nivel de profesionalidad a nuestra plataforma.

De todas formas, cuando hemos creado una página dinámica en PHP lo normal es que tengamos URL no amigables. ¿Cómo las convertimos en amigables de manera automática? Una manera de hacerlo es con un archivo **.htaccess** y gracias a **Apache** y su **mod rewrite**.

Mod rewrite es un módulo de Apache que permite manipular en el servidor las URLs solicitadas. Es decir, cuando llega al servidor la solicitud de una URL, esta se comprueba con respecto a una serie de reglas creadas con Expresiones Regulares creadas para detectar un patrón determinado. Si se encuentra ese patrón en el URL y se cumple la condición impuesta por la Expresión Regular se sustituye el patrón por una cadena de texto o por una acción determinada.

¿Qué vamos a escribir en el archivo **.htaccess**? Como decíamos antes, una serie de Expresiones Regulares para interpretar el URL entrante y crear un URL saliente interpretable por nuestro código. Creamos un archivo de texto, lo grabamos como **.htaccess** y escribimos en él:

- **RewriteEngine on** esta sentencia activa la lectura del archivo porque el servidor lo lee, pero no lo toma en cuenta si no se encuentra activo.
- **Rewrite urls** los comentarios son con almohadilla.
- **RewriteCond** SCRIPT FILENAME !-d y **SCRIPT FILENAME !-f** son instrucciones de permiso que habilitan cambiar la url.
- **RewriteRule** es la sentencia que cambia una regla.

4.5. LESS CSS

Less css es una ampliación a las famosas hojas de estilo CSS, pero a diferencia de éstas funciona como un lenguaje de programación, permitiendo el uso de variables, funciones, operaciones aritmeticas, entre otras, para acelerar y enriquecer los estilos en un sitio web.

Less css no reemplaza a CSS, de hecho el resultado final es una hoja de estilos css completamente funcional y compatible, simplemente ofrece mejoras en el área de desarrollo, por lo que usarlo se vuelve recomendable si quieres ahorrar tiempo de desarrollo, utilizar características avanzadas de estilos y para ahorrarte trabajo de allí su nombre: less css (menos css).

Lo ideal es que todo el código esté en una línea, eso es más rápido y bonito, el código css .min es una sola una línea, sin espacios ni nada y se nota la diferencia, pues esto se carga como dos veces mas rápido que el archivo normal. Cuando mas pesado sea el archivo, mas se nota y claro no se programa sobre esto, se programas en less que es un lenguaje compilador de css junto a un programa que va compilando aparte un archivo css minimizado preparado para el uso en el servidor, lee el min y se trabaja sobre el less y el less permite trabajar mucho mejor el css porque puedes ir metiendo clases dentro de clases.

No es lo mismo que cada clase está pegada a la pared que esto te permite trabajar con hijos y padres, resulta mas útil para programar y optimizar siempre, less te permite eso, programar clases dentro de clases para saber donde está cada cosa.

Less permite compilar varios archivos en uno solo, esto explica que haga las cosas tan rápido pues se guardan ciertas configuraciones en variables. Esto es de gran ayuda porque si tenemos un color principal, uno secundario, etc y luego dos fuentes, no se le llama 465789 que es hexadecimal, lo llamarremos @rojo y se entiendes de forma mas sencilla. Less permite eso, tener un archivo aparte llamado **variables** donde se colocan todas las variables y luego se incluyen en nuestro código less. Cuando se compila y crea el css minimizado, el compilador cambia todas las variables por sus valores esto es @secondaryFont1 se llama a la fuente secundaria donde tenemos nuestras variables.

4.6. Usos de CDN

Una Content Delivery Network (CDN o, en español, una Red de distribución de contenido) es un conjunto de servidores que contienen copias de una misma serie de contenidos (imágenes, vídeos, documentos) y que están ubicados en puntos diversos de una red para poder servir sus contenidos de manera más eficiente.

Esa mayor eficiencia se consigue mediante el balanceo de carga entre los servidores que forman parte del sistema y por su capacidad para servir los contenidos desde el servidor más cercano a cada cliente.

En general, se usan las **Content Delivery Networks** para albergar y distribuir los contenidos estáticos de un sitio web (imágenes, documentos) pero también para la distribución de los contenidos principales de una web (las páginas web propiamente dichas) u otros medios como el vídeo en streaming. En cualquier caso, el objetivo básico que se persigue es siempre el mismo: hacer más eficiente y fiable la distribución de esos contenidos, mediante la eliminación de cuellos de botella y la cercanía al usuario. Además, las CDNs suelen ser gestionadas por terceros, lo que permite a las empresas desentenderse de su creación y mantenimiento, encargándose de eso aquellos cuyo negocio es, específicamente, la distribución de contenidos.

De tal forma, podemos decir que hay dos tipos de links referidos a archivos; los que están hospedados en su hosting, los CDN son mejores porque están en hostings, que han sido asimilados por google entonces, cuando leen esa dirección, carga más rápido el contenido que si el usuario carga el archivo desde el propio hosting. Es un acceso remoto rápido, los buenos frameworks utilizan cdn por eso es más rápido porque se aloja en el propio google. Hay servidores dedicados a cdn el dominio, cuando accedes al dominio empiezas a mandarle instrucciones de lectura de archivos desde el pc al servidor y el servidor se los pide al hosting.

De esta forma, es mas rápido el camino que hay entre el servidor y el hosting de google, que el de estrato, que es nuestro proveedor básicamente, eso es, **cloud download next** son archivos que están ya en la nube a la espera.

En nuestro caso empleamos llamadas a frameworks como **fontawesome** y **semantic**; fontawesome es una extensión para iconos y semantic es una extensión de bootstrap, sin tener que usar bootstrap. Fontawesome que es un script según un correo asociado a cada persona que se registra y proporciona el cdn que apunta a esa dirección para que sea mas rápido el acceso. Se colocan estos links a nuestros css; se declara el tipo de archivo que es `text/css` y así por un archivo de servidor (`htaccess`) se le dice al servidor que le diga al equipo del usuario que se guarde durante X tiempo X archivos. Marcando de esta forma el beneficio de la optimización en nuestra plataforma multimedia.

4.7. Base de datos MySQLi

Aunque no ofrece ninguna novedad respecto a otras interfaces de gestión de bases de datos MySQL mediante PHP, la extensión MySQLi, o como a veces se le conoce, la extensión de MySQL mejorada, viene incluida en las versiones PHP 5 y posteriores y se desarrolló para aprovechar las nuevas funcionalidades que incluyen las versiones de MySQL posteriores a la 4.1.3.

Desde la página oficial de PHP se nos sugiere su uso argumentando que «Esta es la opción recomendada, ya que utilizando el controlador nativo de MySQL resulta en un mejor rendimiento y permite el acceso a funciones que no están disponibles cuando se utiliza la biblioteca de cliente MySQL. Comparada con la extensión MySQL ofrece las siguientes mejoras:

- Interfaz orientada a objetos
- Soporte para Declaraciones Preparadas
- Soporte para Múltiples Declaraciones
- Soporte para Transacciones
- Mejoradas las opciones de depuración
- Soporte para servidor empujado

4.7.1. Base de datos para las categorías

Tal y como se ha explicado en el capítulo anterior, las categorías se podrán crear y destruir desde la "Gestion Web" mediante llamadas a la base de datos cuya estructura es la siguiente:

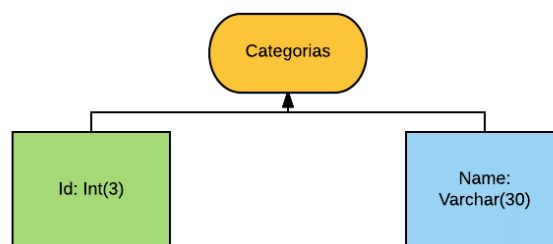


Figura 4.3: Diagrama de categorías en la base de datos

4.7.2. Base de datos para los comentarios

El tratamiento de los comentarios en la publicación de vídeos sigue la siguiente estructura combinatoria en la base de datos:

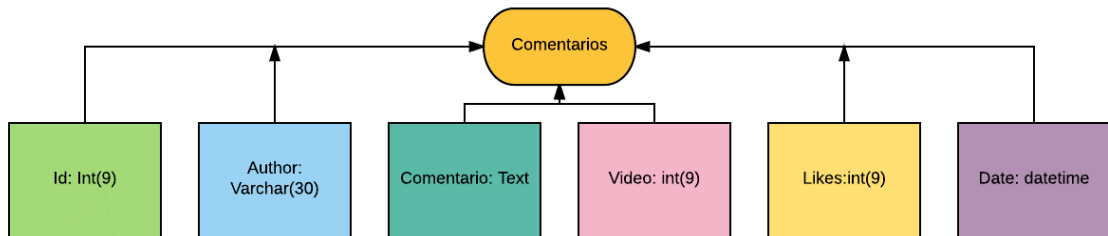


Figura 4.4: Diagrama de comentarios en la base de datos

4.7.3. Base de datos para la gestion web

La configuración de la plataforma en lo que a su gestión y políticas de uso se refiere, corre a cargo del administrador y requiere las siguientes comunicaciones con la base de datos:

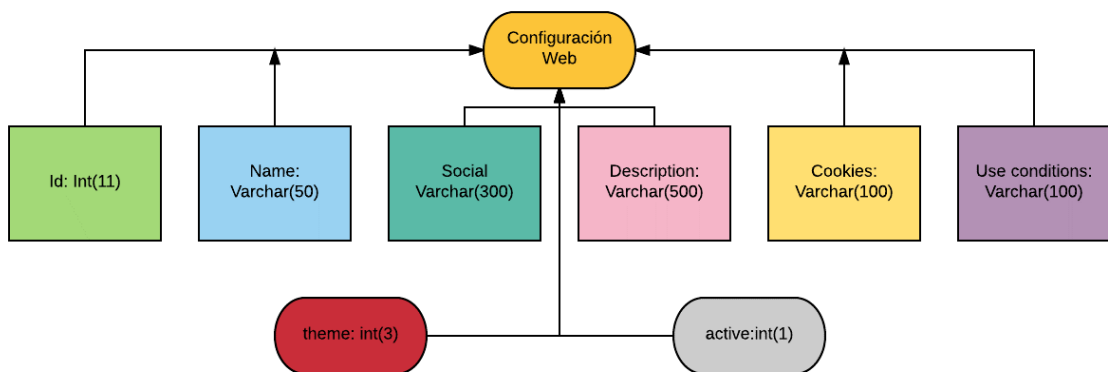


Figura 4.5: Diagrama de gestión web en la base de datos

4.7.4. Base de datos para la gestión de usuarios

La configuración de la plataforma en lo que gestión de los usuario de la misma se refiere, corre a cargo del administrador y requiere las siguientes comunicaciones con la base de datos:

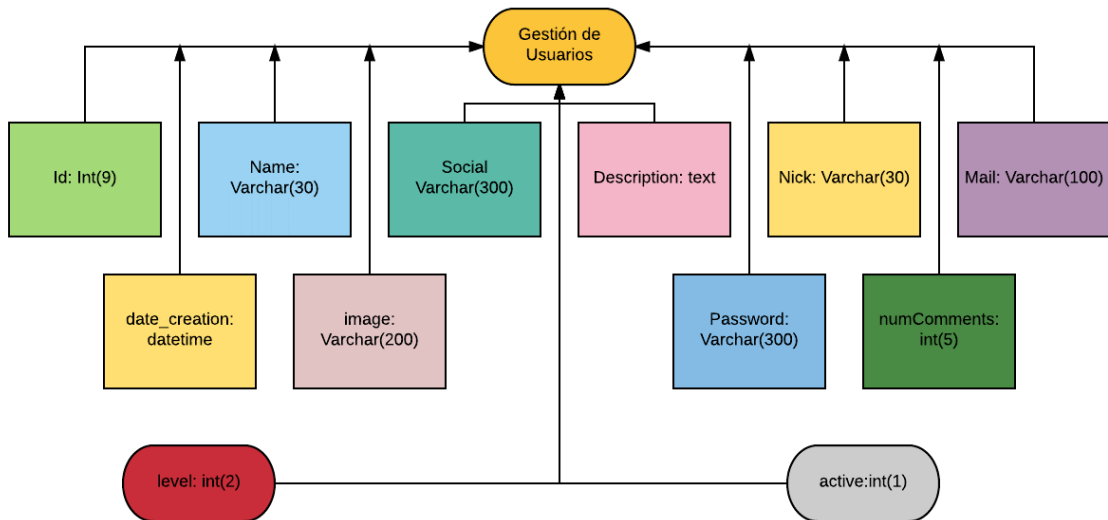


Figura 4.6: Diagrama de gestión de usuarios en la base de datos

4.7.5. Base de datos para la gestión de vídeos

La configuración de la plataforma en lo que gestión y publicación de vídeos , corre a cargo del administrador o moderador y requiere las siguientes comunicaciones con la base de datos:

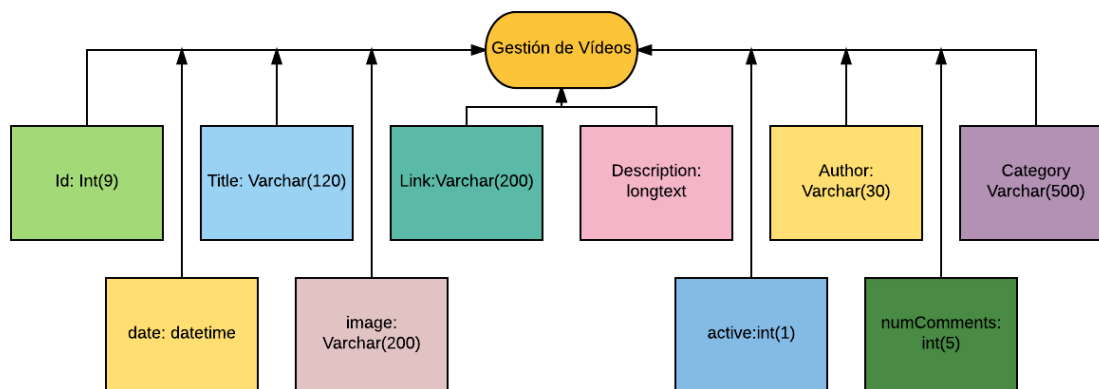


Figura 4.7: Diagrama de gestión de vídeos en la base de datos

Parte III

Parte tercera.

Conclusiones y líneas futuras

Después de todo el desarrollo del proyecto, es pertinente hacer una valoración final del mismo, respecto a los resultados obtenidos, las expectativas o el resultado de la experiencia acumulada. En esta sección se exponen todos esos conceptos y enuncian unas conclusiones finales

Además, considerando también el estado de la técnica, se pueden deducir líneas futuras de trabajo, proponer otros puntos de vista o cualquier otra sugerencia como postámbulo del presente trabajo, para ser considerada por el lector o el tribunal evaluador.

Conclusiones: A continuación, se presentan las conclusiones más importantes que se extraen de la presente proyecto final de carrera:

- El análisis del estado actual de la técnica permite afirmar que las técnicas de programación empleadas en el desarrollo, se han consolidado como una opción competitiva que además, presentan un gran potencial para una más amplia aplicación en el futuro.
- El empleo de diseño responsive es totalmente adecuado y un adelanto a la hora de gestionar y crear elementos web para todo tipo de dispositivos.
- Partir de un diseño de infraestructura y crear un proyecto desde cero es la base perfecta para conseguir el objetivo deseado.
- Emplear diversos lenguajes de programación web y comunicarlos entre ellos proporciona una fuerte herramienta de desarrollo para cualquier tipo de proyecto.
- Un diseño cuidado y una correcta estructura de la base de datos marcan la correcta funcionalidad de comunicación entre los elementos de las bases.

Líneas futuras: El diseño y la creación de elementos web con feed-back usuario-servidor, es un concepto muy extendido a lo largo de los años. Aprender las bases y herramientas para creación de elementos y desarrollar aplicaciones futuras será la base para llegar hasta un trabajo futuro, donde seamos capaces de adaptarnos.

Alfonso Cánovas Baños
13 de septiembre de 2016

Parte IV
Apéndices

Apéndice A

Apéndice

Contenido

A.1	Código fuente .htaces	56
A.2	Código fuente variables.less	60
A.3	Código fuente header.inc	61
A.4	Código fuente managevideos.js	63

Sinopsis

En esta sección se muestran algunos de los códigos fuente empleados en la realización de la plataforma multimedia. Se especifica un código distinto ante cada uno de los lenguajes de programación empleados en el proyecto.

A.1. Código fuente .htaccess

Listing A.1: Código htaccess

```
## Activate Rewrite
RewriteEngine on

## Rewrite urls
RewriteCond %{SCRIPT_FILENAME} !-d
RewriteCond %{SCRIPT_FILENAME} !-f
RewriteRule ^Category/(.+)$ category.php?category=$1
RewriteRule ^Video/(.+)/(.)$ video.php?id=$1
RewriteRule ^Home$ index.php
RewriteRule ^Upload_video$ upload_video.php
RewriteRule ^Manage_videos$ manage_videos.php
RewriteRule ^Manage_video/(.+)/$ manage_video.php?id=$1
RewriteRule ^Remember_password$ remember_password.php

## Add charset
AddDefaultCharset UTF-8

## Rewrite 301
RewriteCond %{HTTP_HOST} ^upctprensa.esy\.es [NC]
RewriteRule ^(.*)$ http://www.upctprensa.esy.es/$1
[L,R=301]

## Specify a Vary: Accept-Encoding header
<IfModule mod_deflate.c>
#The following line is enough for .js and .css
AddOutputFilter DEFLATE js css

#The following line also enables compression by
file content type, for the following list of
Content-Type:s
AddOutputFilterByType DEFLATE text/html text/plain
text/xml application/xml
```

```
#The following lines are to avoid bugs with some
  browsers
BrowserMatch ^Mozilla/4 gzip-only-text/html
BrowserMatch ^Mozilla/4\.0[678] no-gzip
BrowserMatch \bMSIE !no-gzip !gzip-only-text/html
</IfModule>

<IfModule mod_headers.c>
<FilesMatch "\.(js|css|xml|gz)$">
Header append Vary: Accept-Encoding
</FilesMatch>
</IfModule>

## Expires caching
<IfModule mod_expires.c>
ExpiresActive on

# Perhaps better to whitelist expires rules?
  Perhaps.
ExpiresDefault "access
  plus 1 month"

# cache.appcache needs re-requests in FF 3.6 (thx
  Remy ~Introducing HTML5)
ExpiresByType text/cache-manifest "access
  plus 0 seconds"

# your document html
ExpiresByType text/html "access
  plus 0 seconds"

# data
ExpiresByType text/xml "access
  plus 0 seconds"
ExpiresByType application/xml "access
  plus 0 seconds"
ExpiresByType application/json "access
  plus 0 seconds"

# rss feed
```

```

ExpiresByType application/rss+xml      "access
    plus 1 hour"

# favicon (cannot be renamed)
ExpiresByType image/x-icon             "access
    plus 1 month"

# media: images, video, audio
ExpiresByType image/gif                 "access
    plus 3 months"
ExpiresByType image/png                 "access
    plus 3 months"
ExpiresByType image/jpg                 "access
    plus 3 months"
ExpiresByType image/jpeg                 "access
    plus 3 months"
ExpiresByType video/ogg                 "access
    plus 3 months"
ExpiresByType audio/ogg                 "access
    plus 3 months"
ExpiresByType video/mp4                 "access
    plus 3 months"
ExpiresByType video/webm                 "access
    plus 3 months"

# htc files (css3pie)
ExpiresByType text/x-component          "access
    plus 1 month"

# webfonts
ExpiresByType font/truetype              "access
    plus 1 month"
ExpiresByType font/opentype              "access
    plus 1 month"
ExpiresByType application/x-font-woff   "access
    plus 1 month"
ExpiresByType image/svg+xml             "access
    plus 1 month"
ExpiresByType application/vnd.ms-fontobject "
    access plus 1 month"

```



```
# css and javascript
ExpiresByType text/css           "access
    plus 2 months"
ExpiresByType application/javascript "access
    plus 2 months"
ExpiresByType text/javascript    "access
    plus 2 months"

<IfModule mod_headers.c>
Header append Cache-Control "public"
Header set Connection keep-alive
</IfModule>

</IfModule>
```

A.2. Código fuente variables.less

Listing A.2: Código variables.less

```
// ===== LESS VARIABLES =====
// ==== COLORS ====
@baseColor1: #9FD2E3;
@secondColor1: #afe3f4;
@light: #fff;
@dark: #000;

// ==== SOCIAL COLORS ====
@baseColorFacebook: #3b5998;
@secondColorFacebook: #4c6aa9;
@thirdColorFacebook: #2a4887;
@baseColorTwitter: #00aced;
@secondColorTwitter: #11bdfe;
@thirdColorTwitter: #009bdc;
@baseColorGooglePlus: #d34836;
@secondColorGooglePlus: #e45947;
@thirdColorGooglePlus: #c23725;
@baseColorYoutube: #b31217;
@secondColorYoutube: #c42328;
@thirdColorYoutube: #a20106;
@baseColorVimeo: #18b8ec;
@secondColorVimeo: #29c9fd;
@thirdColorVimeo: #07a7db;

// ==== FONTS ====
@baseFont1: 'Raleway', sans-serif;
```

A.3. Código fuente header.inc

Listing A.3: Codigo header.inc

```

<section id="header" data-sr="enter top, hustle 50
  px">
<div class="container box-shadow">
<div class="row">
<!-- Shape info -->
<div id="shape-info" class="col-sm-12 col-md-6">
<!-- Text -->
<p>Lo último</p>

<!-- Buttons -->
<button class="shape-up"><i class="fa fa-chevron-
  left" aria-hidden="true"></i></button>
<button class="shape-down"><i class="fa fa-chevron
  -right" aria-hidden="true"></i></button>

<!-- Shape -->
<div class="ui text shape">
<div class="sides">
<?php
$query = "SELECT id, title, category FROM videos
  WHERE active = '1' ORDER BY id DESC LIMIT 10";
$result = mysqli_query($mysqli, $query);
$cont = 1;
while ($row = mysqli_fetch_array($result)) {
$url_amigable = url_amigable($row['title']);
if ($cont == 1) {
echo "
<a href='Video/".$row['id']."/".$url_amigable."'
  class='ui side active'><span>".$categories[$row[
  'category']]."</span> ".$row['title']."</a>
";
}
}else {
echo "
<a href='Video/".$row['id']."/".$url_amigable."'
  class='ui side'><span>".$categories[$row[
  'category']]."</span> ".$row['title']."</a>
";
}
}

```

```

}

$cont++;
}

?>
</div>
</div>
</div><!-- col-sm-12 col-md-6 -->

<!-- Social and login -->
<div id="social" class="col-sm-12 col-sm-5 col-md-
  offset-1">
<!-- Social -->
<div class="social">
<a href="" class="social-facebook-hover"><i class="
  fa fa-facebook" aria-hidden="true"></i></a>
<a href="" class="social-twitter-hover"><i class="
  fa fa-twitter" aria-hidden="true"></i></a>
<a href="" class="social-google-plus-hover"><i
  class="fa fa-google-plus" aria-hidden="true"></
  i></a>
<a href="" class="social-youtube-hover"><i class="
  fa fa-youtube" aria-hidden="true"></i></a>
<a href="" class="social-vimeo-hover"><i class="fa
  fa-vimeo" aria-hidden="true"></i></a>
</div><!-- social -->

</div><!-- col-sm-12 col-sm-5 col-md-offset-1 -->
</div><!-- row -->
</div><!-- container -->
</section><!-- header -->

<section id="logo" data-sr="enter left, hustle 50
  px">
<div class="container box-shadow">
<div class="row">
<div class="col-sm-12">


```

```


</div><!-- col-sm-12 -->
</div><!-- row -->
</div><!-- container -->
</section><!-- logo -->

```

A.4. Código fuente managevideos.js

Listing A.4: Código managevideos.js

```

$(document).ready(function() {

    // Start

    // Activate or deactivate videos
    $('[data-a="btn-activation"]').click(function
    () {
        var video_id = $(this).attr('data-id-video');
        var video_active = $(this).attr('data-active')
        ;

        var dataString = {
            'video_id'      : video_id,
            'video_active' : video_active
        };
        $.ajax({
            type: "POST",
            url: "php/change_option_active_video.php",
            data: dataString,
            success: function(data) {
                $("#section-incoming").prepend(data);
                if ($('[data-id-video="'+video_id+'"]').
                    hasClass('btn-danger')) {
                    $('[data-id-video="'+video_id+'"]').
                        removeClass('btn-danger').addClass('btn-
                        success');
                }
            }
        });
    });
}

```

```
}else {
$( '[data-id-video="' + video_id + '"' )'.
    removeClass('btn-success').addClass('btn-
    danger');
}

if (video_active == 0) {
$( '[data-id-video="' + video_id + '"' )'.attr('data
    -active', '1');
}else {
$( '[data-id-video="' + video_id + '"' )'.attr('data
    -active', '0');
}

// setTimeout(function() {
//     $(".alert-incoming"+incoming_id).
//     fadeOut(1500);
// },3000);

setTimeout(function() {
$(".alert-incoming").fadeOut(1500);
},3000);
});
});

// DataTable
$( '.dataTable' ).DataTable({
"order": [[ 0, "desc" ]],
"columnDefs": [{ targets: 'no-sort', orderable
: false }],
"lengthMenu": [[25, 50, 100, -1], [25, 50,
100, "Todo"]],
"iDisplayLength": 50
});
} );
```

Bibliografía

- [1] MIKE STEVENSON, «*The ultimate beginners guide to Bootstrap 3.0*»
- [2] ALEX LIBBY, «*Instant LESS CSS Preprocessor How-to*»
- [3] JON DUCKETT, «*JavaScript and JQuery: Interactive Front-End Web Development* »
- [4] JON DUCKETT, «*HTML and CSS: Design and Build Websites*»
- [5] JUAN DIEGO GAUCHAT, «*El Gran Libro De HTML5, CSS3 Y Javascript*»
- [6] ALEXIS GOLDSTEIN, «*HTML5 y CSS3 (Manuales Imprescindibles)* »
- [7] FONT AWESOME GIVES YOU SCALABLE VECTOR ICONS THAT CAN INSTANTLY BE CUSTOMIZED SIZE, COLOR, DROP SHADOW, AND ANYTHING THAT CAN BE DONE WITH THE POWER OF CSS, «<http://fontawesome.io/>»
- [8] W3SCHOOLS IS OPTIMIZED FOR LEARNING, TESTING, AND TRAINING. EXAMPLES MIGHT BE SIMPLIFIED TO IMPROVE READING AND BASIC UNDERSTANDING. TUTORIALS, REFERENCES, AND EXAMPLES ARE CONSTANTLY REVIEWED TO AVOID ERRORS, BUT WE CANNOT WARRANT FULL CORRECTNESS OF ALL CONTENT. , «<http://www.w3schools.com/> »
- [9] BOOTSTRAP MAKES FRONT-END WEB DEVELOPMENT FASTER AND EASIER. IT'S MADE FOR FOLKS OF ALL SKILL LEVELS, DEVICES OF ALL SHAPES, AND PROJECTS OF ALL SIZES., «<http://getbootstrap.com/>»
- [10] JQUERY UI IS A CURATED SET OF USER INTERFACE INTERACTIONS, EFFECTS, WIDGETS, AND THEMES BUILT ON TOP OF THE JQUERY JAVASCRIPT LIBRARY. WHETHER YOU'RE BUILDING HIGHLY INTERACTIVE WEB APPLICATIONS OR YOU JUST NEED TO ADD A DATE PICKER TO A FORM CONTROL, JQUERY UI IS THE PERFECT CHOICE., «<https://jqueryui.com/button/icons>»

- [11] BUILD TOOLS, PERFORMANCE LOGGING, SUPPORT FOR CUSTOM DEFINITIONS, MULTIPLE-LEVELS OF THEME INHERITANCE?A DEVELOPER'S DREAM., *«<http://semantic-ui.com/>»*
- [12] TODO LO QUE TIENES QUE SABER SOBRE .HTACCESS, *«<https://ayudawp.com/todo-sobre-htaccess/>»*
- [13] THE GUIDES AREA IS DESIGNED TO HELP DEVELOPERS LEARN TO BETTER INTERACT WITH THE DATE AND TIME PROBLEM DOMAIN, AND THE MOMENT.JS LIBRARY., *«<http://momentjs.com/>»*