



UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA
Escuela de Ingeniería Informática



Trabajo fin de grado

INSUDOC

ESCUELA DE INGENIERÍA INFORMÁTICA

NOMBRE ALUMNO: **VÍCTOR ALAEZ PADRÓN**

TUTORA: **SONIA MARRERO CÁCERES**

PROFESORA TITULAR UNIVERSIDAD

DEPARTAMENTO DE INFORMÁTICA Y SISTEMAS DE LA ULPGC

LAS PALMAS DE GRAN CANARIA A 11 DE DICIEMBRE DE 2014

Agradecimientos

Deseo dedicar este proyecto a mis familiares y amigos más allegados que han estado apoyándome en todo momento, a mi mujer “Tachu” que ha sido la gran partícipe para poder llegar a este punto y a las que faltan por llegar Julia y Nora.

También me gustaría agradecerle a mi tutora: Sonia Marrero por confiar en mí, por su tiempo perdido y las molestias ocasionadas.

Y en general, agradecer a todas las personas que han intervenido directa o indirectamente en el trabajo y sobre todo en mi vida.

Índice

Introducción.....	5
Capítulo 1.	6
1.1 Estado actual.....	6
1.2 Objetivos	6
1.2.1. Objetivos académicos	6
1.2.2. Objetivos generales del trabajo	7
1.2.3. Competencias	8
Capítulo 2.	10
2.1 Plan de trabajo	10
Capítulo 3.	11
3.1 Aportaciones.....	11
3.2 Viabilidad económica.....	12
Capítulo 4.	18
4.1 Normativa y legislación.....	18
4.2 Licencias software	20
4.3 Requisitos Hardware y software.....	21
Capítulo 5.	26
5.1 Análisis de la aplicación web	26
5.2 Funcionamiento de la aplicación	26
5.3 Desarrollo de Insudoc.....	33
5.3.1. Estructura de la aplicación.....	33
5.3.2. Desarrollo	38
5.4 Metodología de trabajo.....	39
Capítulo 6.	41
6.1 Conclusiones	41
6.2 Trabajo futuro.....	42
6.3 Fuentes de información	43

Introducción

La idea de arrancar con este proyecto surge de casualidad, como casi todas las ideas. Me dedico a la docencia desde hace casi una década, y en mí día a día como tal, una de las cuestiones que más me preocupa es el disponer de tiempo para preparar las clases y los apuntes.

El docente actual, además de impartir clases, dedica gran parte de su tiempo a preparar éstas: elaborar apuntes para el alumnado, realizar documentación de calidad, guías didácticas, exámenes, proyecto docente, etc. Mi idea es aprovechar los recursos TIC e intentar ganar tiempo aunando los procesos de elaboración de apuntes y preparación de las clases.

Una vez terminado los apuntes para los alumnos, muchas veces hay que realizar un resumen y para ello usamos herramientas tales como Powerpoint, Prezi, Macromedia Directors, etc. La idea es que partiendo de los apuntes y gracias a la aplicación podamos generar automáticamente un resumen de los mismos.

Además sabiendo que como yo, hay un gran número de personas en España que se dedican al sector de la formación, creo que es una aplicación que puede servirnos de gran ayuda.

En un estudio de mercado realizado por mí, para conocer las posibilidades de éxito de esta aplicación, encontré un nicho de mercado que podríamos resumir en:

- Sector educativo.
- Oficinas, emprendedores, comunicadores, etc.
- Mercado resegmentado, es decir un mercado que existe pero no he encontrado una herramienta parecido a la que yo quería implementar.

Es una herramienta para profesores, comunicadores y cualquier persona que tenga que trabajar sobre unos apuntes y desee de forma automática generar una presentación.

El nombre de INSUDOC, surge al hacer un juego de letras con las palabras INformática, reSUMen y DOCumentos.

Capítulo 1

1.1 Estado actual

El gran avance tecnológico que han tenido los aplicativos web en las últimas décadas, ha servido para que muchísimos aplicativos y tareas manuales pasen a ser automáticas en el entorno web/ móviles/tablets.

Este trabajo de fin de grado, tiene como objetivo generar una aplicación que ayude a ahorrar tiempo al docente/comunicador/alumno. Con estos cambios que estamos viviendo, viene siendo habitual que cualquier tipo de estructura se convierta en un aplicativo web, como ha pasado con **INSUDOC**. El aspecto positivo del cambio, es la automatización de las tareas para que resulte mucho más sencillo realizarlas y el ahorro de algo tan importante como es de tiempo para que de tal manera podamos dedicarlo a otra cosa quizás más importante.

1.2 Objetivos

1.2. 1 Objetivos Académicos

La meta a conseguir en cualquier educación superior, se rige por la validez de sus estudiantes a la hora de incorporarse al mercado laboral, cobrando especial importancia la asimilación del rápido avance de la tecnología y su constante período de reciclaje, provocando la necesidad de que el alumno se familiarice con métodos, técnicas y tecnologías actuales.

Existe un carácter integrador de conocimientos adquiridos en la titulación como son los obtenidos en las asignaturas:

- Programación
- Ingeniería del software
- Diseño de interfaces humanas
- Prueba y mantenimiento del software
- Diseño y programación de sistemas
- Tecnologías aplicadas a los sistemas de información

1.2.2 Objetivos generales del trabajo

Como ya hemos visto, el principal objetivo del proyecto es el desarrollo de una aplicación web que una el doble esfuerzo de preparar los apuntes y la presentación de estos.

Entre los objetivos generales del proyecto cabe destacar la elección idónea de las herramientas a utilizar, intentando siempre que sean las que están más de actualidad y por supuesto que sean libres.

La finalidad de este trabajo es desarrollar una aplicación web en Node.JS capaz de automatizar la conversión de un documento de texto a una presentación de este.

Esta aplicación sistematizará las tareas convirtiendo un documento de texto, usando los lenguajes de marca, a una presentación dinámica.

Estudiar el funcionamiento de los documentos de textos al guardarlo fue la primera tarea a realizar, una vez entendiendo ésto extrajimos las partes del documento deseadas, para después almacenar y poder mostrarlas en forma de diapositivas en nuestra web.

1.2.3 Competencias

CI101

Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación vigente.

Esta competencia tiene con la elección del hardware y software a utilizar e implementar, especificando por un lado, los motivos de diseño, desarrollo, etc,... explicando exhaustivamente las fases de definición de requisitos, análisis y diseño de los casos de uso necesarios para satisfacer las necesidades de la aplicación web, y por otro lado se justifica la utilización del hardware más adecuado para la implementación de la misma, teniendo en cuenta parámetros de fiabilidad, calidad y seguridad.

En resumen, con esta elección de hardware y software, la implementación de este trabajo de fin de grado da como resultado una aplicación robusta y conforme a los principios éticos y a la legislación vigente.

CI102

Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social.

De todos los elementos de un sistema de gestión de proyectos, la planificación es posiblemente la más importante, puesto que una mala o inexistente planificación conducirá a una mala realización en el proyecto, lo cual repercutirá enormemente tanto en la calidad como en el plazo de terminación del proyecto, ocasionando grandes perjuicios. Esta es la razón por la que esta competencia queda cubierta, puesto que la buena planificación de este trabajo de fin de grado ha dado lugar al propósito fundamental del mismo, que no es más que la finalización en los plazos establecidos así como conseguir de manera adecuada los objetivos propuestos inicialmente. Si además tenemos en cuenta el capítulo Aportaciones, podemos apreciar el impacto del proyecto en el ámbito de aplicación en que nació, o sea, en el Sistema de Garantía y Calidad de la Escuela de Ingeniería Informática, lo que supone de mejora por cuanto se dispondrá de una aplicación que, como consecuencia de su implementación y utilización, aportará una gestión más eficiente a los usuarios del Sistema de Garantía y Calidad. Hay que tener en cuenta que los objetivos que siempre se buscan al desarrollar una aplicación no deben limitarse a resolver el problema sino que hay que intentar elegir la solución más apropiada y conforme a la infraestructura con la que cuenta el cliente.

Esta competencia además queda cubierta en los capítulos de aportación a nuestro entorno socio-económico del tema donde se explica el impacto del proyecto en la economía emergente del sector, en el capítulo de Planificación vemos los pasos en la elaboración del proyecto y la temporización.

CII18

Conocimiento de la normativa y la regulación de la informática en los ámbitos nacional, europeo e internacional (capítulo 3).

TFG01.

Ejercicio original a realizar individualmente y presentar y defender ante un tribunal universitario, consistente en un proyecto en el ámbito de las tecnologías específicas de la Ingeniería en Informática de naturaleza profesional en el que se sinteticen e integren las competencias adquiridas en las enseñanzas

Capítulo 2

2.1 Plan de trabajo

En este capítulo abordamos el plan de trabajo desglosado en las siguientes etapas.

1º Etapa. Documentación. Revisión bibliográfica y análisis previo del proyecto.

Leer documentación sobre las herramientas software libre a utilizar en este proyecto

2ª Etapa. Análisis del proyecto:

- Estudio de un documento de texto xml generado en libreoffice para implementar la aplicación web

3ª Etapa. Diseño del proyecto.

- Estudio del servidor Web Node.js
- Estudio del servidor de base de datos noSQL MongoDB
- Diseño de la base de datos a utilizar

4ª Etapa. Implementación y pruebas

Realización de la aplicación web y una batería de pruebas para comprobar el correcto funcionamiento de la misma.

En esta fase se ha generado el código del diseño realizado y se han construido las bases de datos necesarias. En esta fase el ingeniero de software es el máximo responsable.

5ª Etapa. Memoria

Elaborar la documentación del trabajo, Memoria, CD y más documentación requerida

6ª Etapa. Presentación

Elaboración de la presentación para la exposición y defensa del proyecto

Capítulo 3

3.1 Aportaciones

Con la elaboración de esta herramienta espero cubrir todos los requerimientos necesarios para un mejor aprovechamiento del tiempo de los docentes y comunicadores, y evitar el doble esfuerzo que supone preparar los apuntes y después la presentación de las clases. En la actualidad no he encontrado ninguna herramienta que tenga una funcionalidad parecida, por lo que este proyecto no dispone de esa barrera de entrada.

Lo que se pretende, es unir los documentos de texto con la presentación. Sabiendo que los documentos se pueden guardar en formato XML (lenguajes de marcas) he aprovechado esta funcionalidad y elaborado la herramienta web que se sustenta en el esquema enumerado del documento de texto original. Después, si quisiéramos, únicamente habría que retocar las diapositivas.

INSUDOC pretende aportar una solución a las aplicaciones del docente.

La realización y desarrollo de este sistema brindará beneficios tales como:

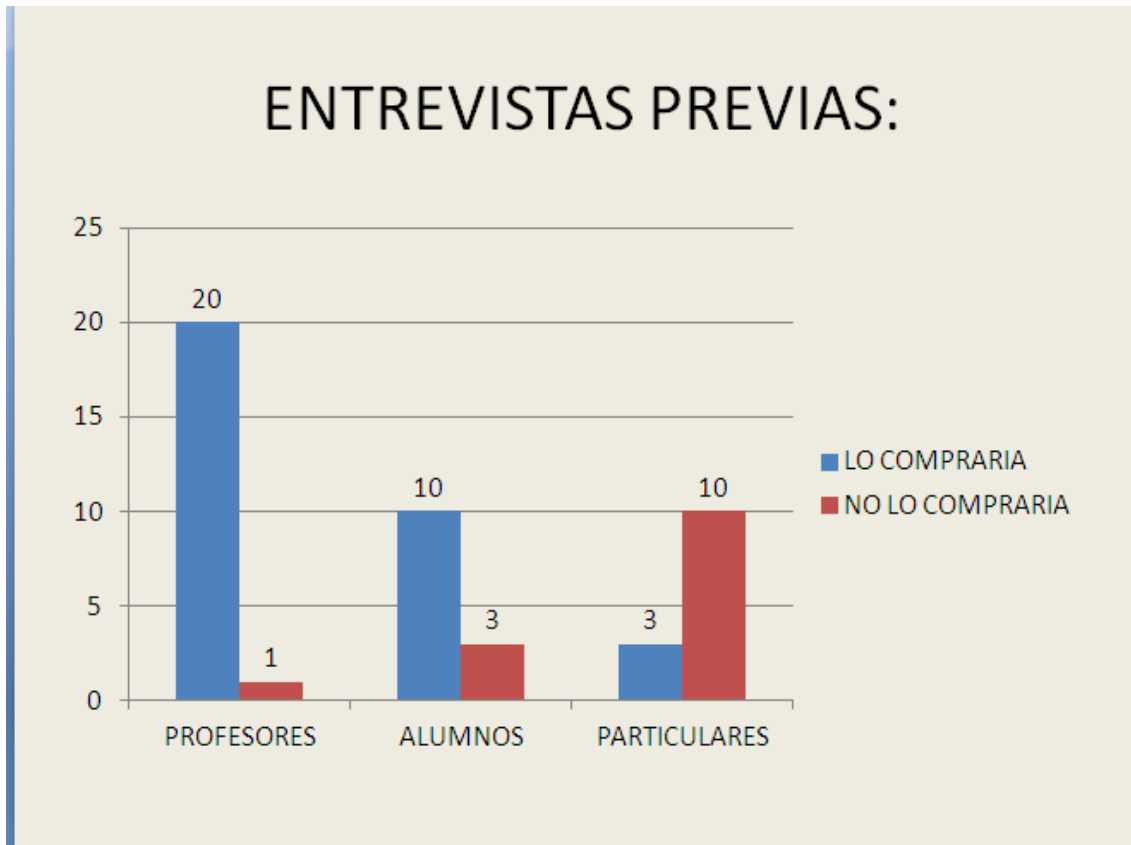
- Ahorro de tiempo y esfuerzo para los profesores.
- Para los alumnos, la preparación de esquema a la hora de estudiar partiendo de sus propios apuntes.
- Evitaría la doble tarea de preparar los apuntes y posteriormente la presentación.

Además sabiendo que actualmente no he encontrado aplicaciones que tengan una funcionalidad parecida, y que no se necesita un gran desembolso económico, considero que es un buen momento para implementar esta aplicación.

3.2 Viabilidad económica

ANÁLISIS PREVIO

He entrevistado a varias personas y la receptividad a este proyecto ha sido bastante alta. Como muestra podemos ver la siguiente estadística realizada a particulares, alumnos y profesores de la ULPGC.



- En España hay 800.000 profesores, 450.000 de enseñanza pública y 350.000 de la enseñanza concertada y privada
- Considero que el mercado potencial de nuestro producto estaría en torno a los 300.000 personas sólo en España, un 22% de profesores más particulares y alumnos.

He analizado también el entorno específico de mi aplicación

ANÁLISIS DEL ENTORNO ESPECÍFICO

COMPETIDORES	PRODUCTOS SUSTITUTIVOS	CLIENTES
<ul style="list-style-type: none"> - Número de competidores -->BAJO - Ritmo de crecimiento en el sector -->ALTA - Diferenciación del producto -->ALTA - Costes de cambio del producto (para el usuario) -->BAJO 	<ul style="list-style-type: none"> • Disponibilidad de sustitutos próximos -->REAL • Propensión de los compradores a la sustitución --> ALTA • Costes de cambio (para el usuario) -->Bajo • Relación calidad/precio de los sustitutos -->EQUILIBRADO • Obsolescencia que el sustituto origina en el sector -->BAJA 	<ul style="list-style-type: none"> • Grado de concentración de los clientes →Medio • Volumen de compra de los clientes → Medio • Disponibilidad de sustitutos para los productos del sector →BAJA • Costes de cambio para el cliente →BAJO • Amenaza de integración hacia atrás del cliente →BAJO • Amenaza de integración hacia delante del sector →Alta

DEBILIDADES Y FORTALEZAS DE INSUDOC

DEBILIDADES	FORTALEZAS
<ul style="list-style-type: none"> • Dependencia de integración a otras tecnologías • Desconocimiento de desarrollo software y marketing que conlleva • La situación geográfica de arranque del negocio. • Falta de nombre por ser una nueva aplicación 	<ul style="list-style-type: none"> • Producto diferenciado. • Bajo coste de producción • Producto totalmente innovador • Experiencia y formación de los desarrolladores en el campo Informático, tecnológico y Marketing • Experiencia en el mundo empresarial

ALIADOS

- Nuestros principales aliados serian las aplicaciones ofimáticas de software libre
- En un principio al tratarse de software libre, no vemos ningún riesgos

RECURSOS HUMANOS NECESARIOS

Puesto: Responsable de proyecto/Programador

Tareas

- Gestión de proyecto
- Desarrollo del proyecto
- Supervisión del proyecto
- Marketing inicial del proyecto

Nº Personas: 1

Horas de Trabajo

- 1) 40 horas semanales en jornadas de 8 horas/día por trabajador.

Retribución: 1400 € (salario bruto repartido en 12 pagas).

Puesto: Responsable Marketing y Ventas.

Tareas

- Gestión del diseño y producción de publicidad.
- Gestión del diseño y puesta en marcha de otros métodos de captación de clientes.
- Gestión de la calidad.
- Supervisión de productividad del departamento.
- Comercial

Nº Personas: 1

Horas de Trabajo

- 1) 40 horas semanales en jornadas de 8 horas/día por trabajador.

Retribución: 1400 € (salario bruto repartido en 12 pagas)

FINANCIACION INICIAL

2 FINANCIACIÓN NECESARIA INSUMDOC

FINANCIACIÓN	DESEMBOLSO	%
RECURSOS PROPIOS	5.005,0	66,7%
Capital	3.005,0	40,0%
Aportaciones no dinerarias	2.000,0	26,6%
Préstamos de socios a la sociedad	0,0	0,0%
RECURSOS AJENOS	2.500,0	33,3%
Préstamos bancarios	0,0	0,0%
Leasing	0,0	0,0%
Subvenciones	2.500,0	33,3%
Otros	0,0	0,0%
OTRAS FUENTES	0,0	0,0%
Capital riesgo y otras fuentes	0,0	0,0%
TOTAL	7.505,0	100,0%
Total a financiar	5.800,00	

COSTES FINANCIEROS LOS 5 PRIMEROS AÑOS

The screenshot shows a Microsoft Excel spreadsheet titled 'Plarbita económico-insudoc [Modo de compatibilidad] - Microsoft Excel'. The spreadsheet is divided into several sections:

- 4 CUENTA DE RESULTADOS PREVISIONAL**: A table showing financial data from 2015 to 2019. It includes 'Ingresos' (Total ingresos: 4.000,00 to 441.000,00) and 'Gastos' (Total gastos: 47.285,00 to 72.968,55). It also shows 'Beneficios o Pérdidas' and 'Bañ'.
- ESTIMACIÓN DE GASTOS FIJOS (1ER AÑO)**: A single value of 47.285,00.
- ESTIMACIÓN DE VENTAS**: A table with columns for 'PRODUCTOS / SERVICIOS', 'UNIDADES', 'PRECIO UNITARIO', 'IMPORTE', 'COSTE UNITARIO', 'COSTE TOTAL', and '% MARGEN'.

El primer año, al ser un producto nuevo las previsiones de ingreso no son muy altas.

A partir del 2016, empezaríamos a ver la rentabilidad en el negocio y además esta sería escalable

Como se puede comprobar en Excel, es a partir del tercer año de vida de la empresa (2017), cuando comenzamos a obtener beneficios y la rentabilidad es positiva, alcanzándose de esta forma la viabilidad del proyecto.

Al realizar el estudio, cuantificando los recursos necesarios y la simulación correspondiente de lo que supondría su puesta en marcha, podemos concluir que el proyecto es viable. He identificado datos relevantes que nos invitan a pensar en esa viabilidad del proyecto:

- El proyecto será rentable a medio plazo, ya que los dos primeros años estaremos recuperándonos del esfuerzo inicial realizado
- Inicialmente partimos de una inversión por parte de los socios de 3006 € para montar la sociedad limitada además de un préstamo por parte de los socios de 28.000 para pagar sus propios sueldos el primer año
- A partir del segundo año empezarían a crecer las ventas, una vez consigamos salir al mercado nacional e internacional.
- Unos de los riesgos que vemos en nuestra insularidad y estar alejado, desde un punto de vista físico, del mercado nacional e internacional

LIENZO DE MODELO NEGOCIO

Empresa INSUDOC				
<p> Aliados/<u>Partners</u> clave</p> <ul style="list-style-type: none"> • ULPGC • Institutos • Colegios • Gabinetes 	<p> Actividades clave</p> <ul style="list-style-type: none"> • Marketing • Mejora constante del producto • Ventajas que aporta <p> Recursos clave</p> <ul style="list-style-type: none"> • Desarrolladores especializados 	<p> Propuesta de valor</p> <p>Par los profesores/comunicadores que deseen resumir de una forma rápida los apuntes dados al alumno <u>Insumdoc</u> permite realizarla al instante</p> <p>Al contrario que muchas herramientas que te obligan a volver a escribir todo de nuevo <u>Insumdoc</u> lo hace de forma automática</p>	<p> Relación con el cliente</p> <ul style="list-style-type: none"> • Visitas a universidades, institutos, etc • Convenciones Tecnológicas • A través de <u>internet</u> • Redes sociales, web, etc <p> Canales</p> <ul style="list-style-type: none"> • Canal virtual mayoritariamente • Canal físico (cd, pendrive) • <u>Appstore</u> 	<p> Segmentos de cliente</p> <ul style="list-style-type: none"> • Profesores • Estudiantes • Asesores • Comunicadores • Cualquier persona que tenga la necesidad de resumir un documento de texto en un resumen/Presentación
<p> Estructura de costes</p> <ul style="list-style-type: none"> • Fuerte inversión en Marketing • Directivos serán los primeros trabajadores • Local propio • Desarrollo 		<p> Flujos de ingresos</p> <ul style="list-style-type: none"> • Venta por canales de distribuciones • Suscripciones al producto • Actualizaciones del producto • Publicidad en la Web • <u>Multilicencias</u> 		

Capítulo 4

4.1 Normativa y legislación

Un análisis de las legislaciones que se han promulgado en diversos países arroja que las normas jurídicas que se han puesto en vigor están dirigidas a proteger la utilización abusiva de la información reunida y procesada mediante el uso de computadoras, e incluso en algunas de ellas se ha previsto formar órganos especializados que protejan los derechos de los ciudadanos amenazados por los ordenadores.

Desde hace aproximadamente diez años la mayoría de los países europeos han hecho todo lo posible para incluir dentro de la ley, la conducta punible penalmente como el acceso ilegal a sistemas de computo o el mantenimiento ilegal de tales accesos, la difusión de virus o la interceptación de mensajes informáticos.

En la mayoría de las naciones occidentales existen normas similares a los países europeos. Todos estos enfoques están inspirados por la misma de preocupación de contar con comunicaciones electrónicas, transacciones e intercambios tan confiables y seguros como sea posible.

Dar un concepto sobre delitos informáticos no es una labor fácil y esto es en razón de que su misma denominación alude a una situación muy especial, ya que para hablar de "delitos" en el sentido de acciones tipificadas o contempladas en textos jurídico penales, se requiere que la expresión "delitos informáticos" este consignada en los códigos penales, lo cual en nuestro país, al igual que en muchos otros, no ha sido objeto de tipificación aún; sin embargo, muchos especialistas en derecho informático emplean esta alusión a los efectos de una mejor conceptualización

Normativa y regulación de la informática en el ámbito europeo.

Hasta ahora, el principal esfuerzo europeo por regular el tema de los delitos informáticos dio como resultado el "Convenio sobre la Ciberdelincuencia", de 21 de noviembre de 2001. Este documento fue firmado por los representantes de cada país miembro del Consejo de Europa, aunque su eficacia depende de su posterior refrendo por los órganos nacionales de cada país firmante.

El "Convenio sobre la Ciberdelincuencia" permitió la definición de los delitos informáticos y algunos elementos relacionados con éstos, tales como "sistemas informáticos", "datos informáticos", o "proveedor de servicios".

Estos delitos informáticos fueron clasificados en cuatro grupos:

1º Delitos contra la confidencialidad, la integridad y la disponibilidad de los datos y sistemas informáticos.

- Acceso ilícito a sistemas informáticos.
- Interceptación ilícita de datos informáticos.
- Interferencia en el sistema mediante la introducción, transmisión, provocación de daños, borrado, alteración o supresión e estos.
- Abuso de dispositivos que faciliten la comisión de delitos

2º Delitos informáticos.

- Falsificación informática que produzca la alteración, borrado o supresión de datos informático que ocasionen datos no auténticos.
- Fraudes informáticos.

3º Delitos relacionados con el contenido.

- Delitos relacionados con la pornografía infantil.

4º Delitos relacionados con infracciones de la propiedad intelectual y derechos afines.

Conviene destacar que en el "Convenio sobre la Ciberdelincuencia" se encomienda a cada Parte que tome las medidas necesarias para tipificar como delito en su derecho interno cada uno de los apartados descritos en cada categoría.

En la Disposición 14221 del BOE núm. 226 de 2010, encontramos el Instrumento de Ratificación del Convenio sobre la Ciberdelincuencia, hecho en Budapest el 23 de noviembre de 2001.

Normativa y regulación de la informática en el ámbito nacional.

Leyes y Decretos Ley.

Ley Orgánica de Protección de datos de carácter personal: régimen sancionador aplicable (BOE no298 de 14/XII/99 que publicó la Ley Org. 15/1999 de 13 de Dic.)

Objeto: Proteger y garantizar las libertades públicas y derechos fundamentales de las personas, especialmente su HONOR e INTIMIDAD personal y familiar.

Aspectos de interés: Serán responsables: "Los responsables de los ficheros o de los tratamientos" y "los encargados de los tratamientos"

Tipos de Infracciones:

- Leves (art.44.2): multas de 100.000 a 10M pts.
- Ej. Rectificar datos o no comunicarlos a la Agencia de Protección Datos.
- Graves (art.43): multas de 10M a 50Mpts:
- No mantener sistemas de seguridad, obstrucción a inspección, uso en provecho propio...
- Muy Graves (art.45): multas de 50M a 100M ("Conductas reprochables")
- Ej. Vulnerar a propósito el secretismo, etc...
-

Ley 7/1998 de 13 de Abril que regula las condiciones generales de contratación. R.D. 1906/1999 de 17/XII que regula la contratación telefónica. R.D.Ley 14/1999 de 17/XII sobre Firma Electrónica (BOE No224 de 18/XII)

Firma electrónica: Dispositivo electrónico que permite la identificación del signatario de las operaciones realizadas por Internet.

Identifica: El firmante (autenticación) y Evita el retracto (no repudio).

Código Penal

Ley Orgánica 10/1995 de 23/XI

Tipifica delitos y faltas por el uso de la informática, concretamente contra la Intimidad, Patrimonio, Socioeconómicos y Propiedad Intelectual.

Título X: "Delitos contra la intimidad, derecho a la propia imagen y la inviolabilidad del Domicilio".

Apoderarse de papeles, e-mails, mensajes, otros...

4.2 Licencia software

En este proyecto sólo hemos usado software y herramientas con licencia de software Libre que se definen a continuación

GNU Lesser General Public License

La Licencia Pública General Reducida de GNU, o más conocida por su nombre en inglés GNU Lesser General Public License (antes GNU Library General Public License o Licencia Pública General para Bibliotecas de GNU), o simplemente por su acrónimo del inglés GNU LGPL, es una licencia de software creada por la Free Software Foundation que pretende garantizar la libertad de compartir y modificar el software cubierto por ella, asegurando que el software es libre para todos sus usuarios.

Esta licencia permisiva se aplica a cualquier programa o trabajo que contenga una nota puesta por el propietario de los derechos del trabajo estableciendo que su trabajo puede ser distribuido bajo los términos de esta "LGPL Lesser General Public License". El "Programa", utilizado en lo subsecuente, se refiere a cualquier programa o trabajo original, y el "trabajo basado en el Programa" significa ya sea el programa o cualquier trabajo derivado del mismo bajo la ley de derechos de autor: es decir, un trabajo que contenga el Programa o alguna porción de él, ya sea íntegra o con modificaciones o traducciones a otros idiomas.

Licencia MIT

La licencia MIT es una de tantas licencias de software que ha empleado el Instituto Tecnológico de Massachusetts (MIT, Massachusetts Institute of Technology) a lo largo de su historia, y quizás debería llamarse más correctamente licencia X11, ya que es la licencia que llevaba este software de muestra de la información de manera gráfica X Window System originario del MIT en los años 1980. Pero ya sea como MIT o X11, su texto es idéntico.

El texto de la licencia no tiene copyright, lo que permite su modificación. No obstante esto, puede no ser recomendable e incluso muchas veces dentro del movimiento del software de código abierto desaconsejan el uso de este texto para una licencia, a no ser que se indique que es una modificación, y no la versión original.

La licencia MIT es muy parecida a la licencia BSD en cuanto a efectos

4.3 Requisitos Hardware y software

Durante el desarrollo del proyecto se ha utilizado una serie de recursos para llevar a cabo los objetivos. Los recursos los podemos distinguir en dos grupos principales: hardware y software

REQUISITOS HARDWARE

Los requisitos hardware para el desarrollo de este trabajo son un servidor, que en este caso será un servidor local implementado en mi ordenador personal donde se aloja la aplicación web

REQUISITOS SOFTWARE

UBUNTU

Ubuntu es un sistema operativo basado en Linux y que se distribuye como software libre, el cual incluye su propio entorno de escritorio denominado Unity. Su nombre proviene de la ética homónima, en la que se habla de la existencia de uno mismo como cooperación de los demás.

Está orientado al usuario novel y promedio, con un fuerte enfoque en la facilidad de uso y en mejorar la experiencia de usuario. Está compuesto de múltiple software normalmente distribuido bajo una licencia libre o de código abierto. Estadísticas web sugieren que la cuota de mercado de Ubuntu dentro de las distribuciones Linux es, aproximadamente, del 49%,^{3 4} y con una tendencia a aumentar como servidor web.⁵ y un importante incremento activo de 20 millones de usuarios para fines del 2011.⁶

Su patrocinador, Canonical, es una compañía británica propiedad del empresario sudafricano Mark Shuttleworth. Ofrece el sistema de manera gratuita, y se financia por medio de servicios vinculados al sistema operativo^{7 8} y vendiendo soporte técnico.⁹ Además, al mantenerlo libre y gratuito, la empresa es capaz de aprovechar los desarrolladores de la comunidad para mejorar los componentes de su sistema operativo.

LIBRE OFFICE

LibreOffice es una suite ofimática libre y de código abierto desarrollada por The Document Foundation. Se creó como bifurcación de OpenOffice.org en 2010.

Cuenta con un procesador de texto (Writer), un editor de hojas de cálculo (Calc), un gestor de presentaciones (Impress), un gestor de bases de datos (Base), un editor de gráficos vectoriales (Draw) y un editor de fórmulas matemáticas (Math).

Está diseñada para ser compatible con las principales suites ofimáticas, incluyendo Microsoft Office, aunque algunas características de diseño y atributos de formato son manejados de forma diferente o no son compatibles.⁵ LibreOffice está disponible en más de 120 idiomas incluyendo español, catalán, vasco, gallego⁶ y para diferentes sistemas operativos,⁷ incluyendo Microsoft Windows, Mac OS X 10.4 Tiger o superior y GNU/Linux.

HTML 5

HTML5 (HyperText Markup Language, versión 5) es la quinta revisión importante del lenguaje básico de la World Wide Web, HTML. HTML5 especifica dos variantes de sintaxis para HTML: un «clásico» HTML (text/html), la variante conocida como HTML5 y una variante XHTML conocida como sintaxis XHTML5 que deberá ser servida como XML.^{1 2} Esta es la primera vez que HTML y XHTML se han desarrollado en paralelo.

La versión definitiva de la quinta revisión del estándar se publicó en octubre de 2014.³

Al no ser reconocido en viejas versiones de navegadores por sus nuevas etiquetas, se recomienda al usuario común actualizar a la versión más nueva, para poder disfrutar de todo el potencial que provee HTML5.

XML (eXtensible Markup Language)

El Lenguaje de Etiquetado de Documentos eXtensible (XML, por sus siglas en inglés, eXtensible Markup Language) es un metalenguaje de marcas desarrollado para definir lenguajes para diferentes necesidades. Así, su aplicación no sólo se encuentra en Internet, sino que se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. Se puede usar en bases de datos, editores de texto, hojas de cálculo, etc.

XML es una tecnología sencilla con un papel muy importante en la actualidad, ya que permite la compatibilidad entre sistemas para compartir la información de una manera segura, fiable y fácil.

Con la creación de XML se buscaba cumplir varios objetivos (cumplidos finalmente):

- que fuera idéntico a la hora de enviar, recibir y procesar la información en HTML para aprovechar toda la tecnología implantada de éste;
- que fuera conciso desde el punto de vista de los datos y la manera de guardarlos;
- que fuera extensible para poder utilizarse en todos los campos del conocimiento;
- que fuese fácil de leer, editar e implementar; y
- que fuese fácil de implantar, programar y aplicar a los distintos sistema

JAVASCRIPT

JavaScript es un lenguaje interpretado orientado a páginas web. Se ejecuta en el lado del cliente, lo que significa que es el propio navegador quien soporta la carga de procesamiento. Fue desarrollado por Netscape Communications Corporation para su Navigator 2.0.

JavaScript permite mejorar la gestión cliente/servidor. Un guión de JavaScript puede tratar y gestionar localmente, en el cliente (navegador del usuario), eventos tales como:

- comprobar la validez de los campos cumplimentados en un formulario,
- abrir y cerrar ventanas,
- cambios dinámicos en una página (aspectos y contenidos),
- cadenas de texto, y
- operaciones aritméticas.

También puede utilizarse para realizar varias tareas a la vez, como abrir una página web y simultáneamente visualizar un video, reproducir sonido o ejecutar un applet de Java. Dado que su cometido es ampliar el HTML, JavaScript es un lenguaje con algunas limitaciones que, indirectamente, confieren seguridad para el navegante. Así, por ejemplo, JavaScript no puede (no sabe) crear, leer, modificar o borrar ficheros del usuario, con una excepción, las cookies. Carece también de mecanismos para establecer conexiones de red, llegando como mucho a ser capaz de enviar correos o formularios.

Tradicionalmente utilizado en páginas web HTML, para realizar tareas y operaciones en el marco de la aplicación cliente/servidor, JavaScript se ha convertido en un verdadero lenguaje de programación que aporta la potencia de cálculo del navegador para aumentar la usabilidad de aplicaciones web con técnicas avanzadas como AJAX.

JavaScript se diseñó con una sintaxis similar al C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo Java y JavaScript no están relacionados y tienen semánticas y propósitos diferentes.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

Ejemplo de código JavaScript incrustado en código HTML y que permite visualizar

texto en el navegador, además de abrir una pequeña ventana informativa, es el siguiente:

```
<html>
<head>
<title>Ejemplo JavaScript</title>
</head>
<body>
<script>
<!--document.write("Esto es un ejemplo de JavaScript");
alert("Bienvenid@ a JavaScript");
//-->
</script>
</body>
</html>
```

JQUERY

jQuery es una biblioteca de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web. Fue presentada el 14 de enero de 2006 en el BarCamp NYC. jQuery es la biblioteca de JavaScript más utilizada.¹

jQuery es software libre y de código abierto, posee un doble licenciamiento bajo la Licencia MIT y la Licencia Pública General de GNU v2, permitiendo su uso en proyectos libres y privados.² jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

CSS

Hoja de estilo en cascada o CSS (siglas en inglés de cascading style sheets) es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML2 (y por extensión en XHTML). El World Wide Web Consortium (W3C) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario navegadores.

La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación.

La información de estilo puede ser definida en un documento separado o en el mismo documento HTML. En este último caso podrían definirse estilos generales en la cabecera del documento o en cada etiqueta particular mediante el atributo «style

NODE.JS

Node.js es un entorno de programación en la capa del servidor basado en el lenguaje de programación ECMAScript, asíncrono, con I/O de datos en una arquitectura orientada a eventos y basado en el motor V8 de Google. Fue creado con el enfoque de ser útil en la creación de programas de red altamente escalables, como por ejemplo, servidores web.² Fue creado por Ryan Dahl en 2009 y su evolución está apadrinada por la empresa Joyent, que además tiene contratado a Dahl en plantilla.^{3 4}

Node.js es similar en su propósito a Twisted o Tornado de Python, Perl Object Environment de Perl, React de PHP, libevent o libev de C, EventMachine de Ruby, vibe. d de D y de Java existe Apache MINA, Netty, Akka, Vert.x, Grizzly o Xsocket. Al contrario que la mayoría del código JavaScript, no se ejecuta en un navegador, sino en el servidor. Node.js implementa algunas especificaciones de CommonJS.⁵ Node.js incluye un entorno REPL para depuración interactiva

EXPRESS.JS

Express es sin duda el framework más conocido de node.js, es una extensión del poderoso connect y esta inspirado en Sinatra, además es robusto, rápido, flexible, simple...

Sin duda el éxito de express radica en lo sencillo que es usarlo, y además abarca un sin número de aspectos que muchos desconocen pero son necesarios.

De entre las tantas cosas que tiene este framework podemos destacar:

- Session Handler
- 11 middleware poderosos así como de terceros.
- cookieParser, bodyParser...
- vhost
- router, y contando

MONGODB

MongoDB (de la palabra en inglés “humongous” que significa enorme) es un sistema de base de datos NoSQL orientado a documentos, desarrollado bajo el concepto de código abierto.

MongoDB forma parte de la nueva familia de sistemas de base de datos NoSQL. En vez de guardar los datos en tablas como se hace en las bases de datos relacionales, MongoDB guarda estructuras de datos en documentos tipo JSON con un esquema dinámico (MongoDB llama ese formato BSON), haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida.

El desarrollo de MongoDB empezó en octubre de 2007 por la compañía de software 10gen. Ahora MongoDB es una base de datos lista para la producción de uso y con muchas características (features). Esta base de datos es altamente utilizada en las industrias¹ y MTV Network,² Craigslist³ y Foursquare⁴ son algunas de las empresas que utilizan esta base de datos

Capítulo 5

5.1 Análisis de la aplicación web

Aplicación web que de manera automática genere una presentación resumida del contenido de un documento de texto. Se utilizará para ello un lenguaje de programación que, aplicando técnicas de reconocimiento automático de patrones para la identificación de estructuras de textos, genere la presentación.

5.2 Funcionamiento de la aplicación

La pantalla principal de la aplicación es la que se puede ver en la Figura 1. En esta pantalla se podrán ver los cursos que se han creado, si existen, y un enlace para ir al formulario donde se pueden crear cursos nuevos así como un buscador para poder buscar un curso de la plataforma.

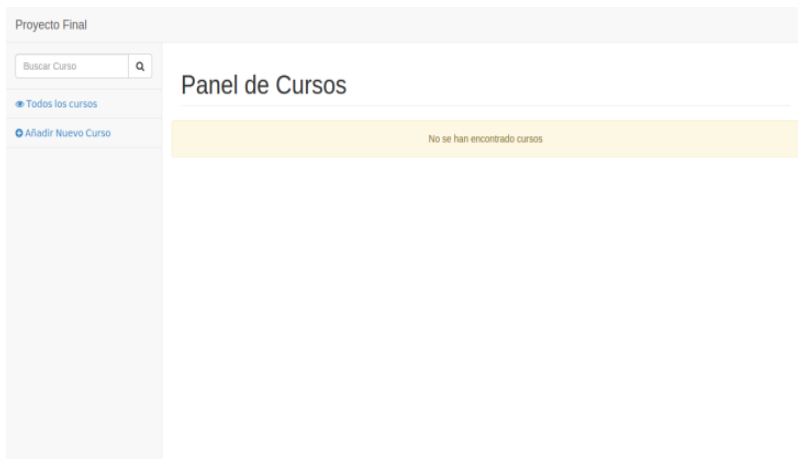


Figura 1

Si queremos añadir un nuevo curso se mostrará la pantalla de la figura 2. En ella se nos muestra un formulario donde todos los campos son obligatorios y donde anexaremos el documento de texto que queremos elaborar la presentación. Se pondrán introducir tantos documentos como temas tenga el curso.

Proyecto Final

Buscar Curso

Todos los cursos

Añadir Nuevo Curso

Administración del curso

Ficha del Curso

Definición del curso

Título de Curso

¿Cuántos temas tiene el curso?

1

Título para el tema 1

Fichero para el tema 1

Seleccionar archivo Ningún archivo seleccionado

Guardar

Figura 2

Una vez pulsado en el botón guardar (figura 2) se mostrar la imagen de la figura 3, con lo que el curso se habrá introducido correctamente en la base de datos.

Proyecto Final

Buscar Curso

Todos los cursos

Añadir Nuevo Curso

Administración del curso

Ficha del Curso

Definición del curso

Título de Curso

Curso Municipios de Las Palmas

¿Cuántos temas tiene el curso?

1

Título para el tema 1

Municipios de Las Palmas

Fichero para el tema 1

Seleccionar archivo Prueba munic...nuevo_1.xml

Guardar

Información

Curso creado correctamente


Aceptar

Figura 3

Ya creado el curso, veremos la pantalla de la figura 4, en ella podemos observar tres menús tipos pestaña. En la primera pestaña aparece la definición del curso, en la segunda pestaña, cada uno de los temas del curso, (mínimo uno) y en la última, se podrá acceder a la presentación de cada uno de los temas del curso.

The screenshot shows a web interface for course management. At the top left, it says 'Proyecto Final'. Below that is a search bar labeled 'Buscar Curso' with a magnifying glass icon. To the left of the main content area is a sidebar with two menu items: 'Todos los cursos' and 'Añadir Nuevo Curso'. The main content area is titled 'Administración del curso' and contains a 'Ficha del Curso' section. This section has three tabs: 'Definición del curso' (selected), 'Temas del Curso', and 'Diapositivas del Curso'. Under the 'Definición del curso' tab, there are several input fields: 'Título de Curso' with the value 'Curso Municipios de Las Palmas', '¿Cuántos temas tiene el curso?' with the value '1', 'Título para el tema 1' with the value 'Municipios de Las Palmas', and 'Fichero para el tema 1' with the value 'Prueba municipios_nuevo_1.xml'.

Figura 4

En la figura 5 podemos ver un ejemplo de cómo se mostraría un tema enlazado con nuestra aplicación web. En ella podemos definir el tamaño de cada diapositiva, marcando el inicio y fin de éstas. Además pulsando sobre el icono  se mostrará un cuadro de texto (Figura 6) donde podemos introducir más argumentos para nuestra presentación.

Proyecto Final

Buscar Curso

Todos los cursos

Añadir Nuevo Curso

Administración del curso


Ficha del Curso

Definición del curso | Temas del Curso | Diapositivas del Curso

Municipios de Las Palmas

Generar Diapositiva


1. Gran Canaria + Definir diapositiva



1. Agaete + Definir diapositiva

2. Agüimes + Definir diapositiva

3. Artenara + Definir diapositiva



4. Arucas + Definir diapositiva

Figura 5

Proyecto Final

Buscar Curso

Todos los cursos

Añadir Nuevo Curso

Administración del curso

Definición del curso | Temas del Curso | Diapositivas del Curso

Municipios de Las Palmas

Generar Diapositiva

1. Gran Canaria -

Ejemplo de comentario que aparecería en la diapositiva

Guardar

Definir diapositiva




Figura 6

Una vez guardado el comentario, el aspecto que tendrá este será el de la figura 7.



Figura 7

Definido el contenido de las diapositivas (figura 8) y creados los comentarios para cada punto de las diapositivas, se deberá pulsar sobre el botón generar diapositiva.

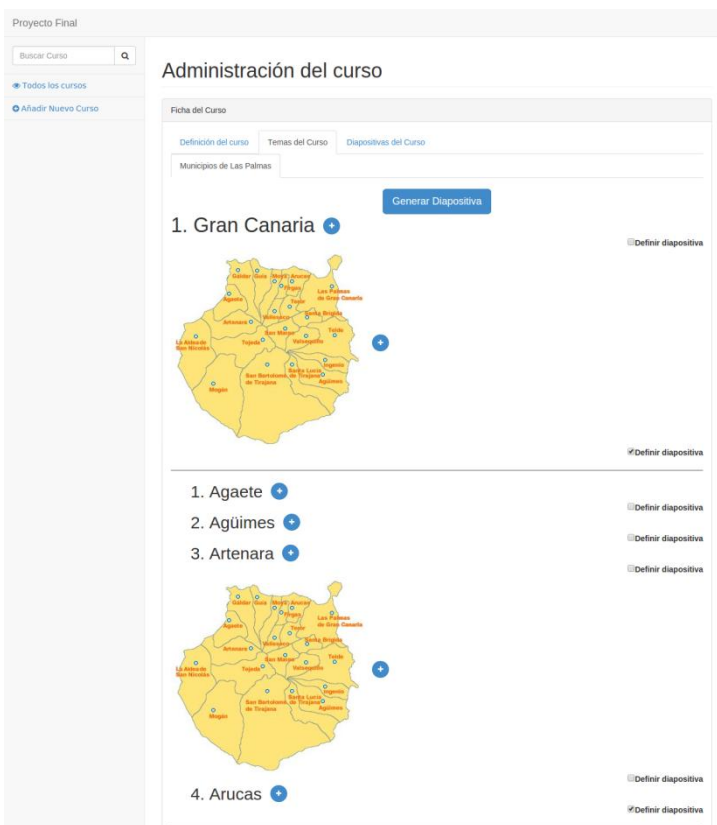


Figura 8

Generada la diapositiva se mostrará la imagen de la figura 9 informándonos que todo ha salido correctamente

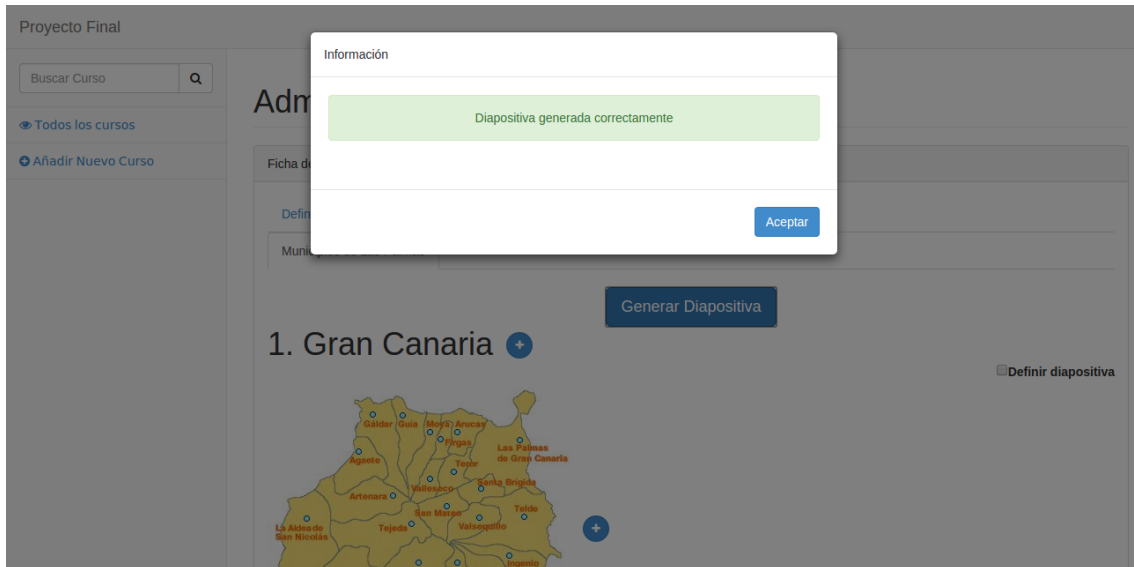


Figura 9

Finalmente accediendo a la pestaña “diapositiva del curso” (figura 10) y clicando sobre el tema del curso que acabamos de generar se nos mostrará la presentación de ese tema (figura 11 y 12).



Figura 10

Curso Municipios de Las Palmas

Municipios de Las Palmas

1/

Figura 11

1. Gran Canaria



2/

Figura 12

En la última diapositiva del tema se mostrara la diapositiva de la figura 13, con un botón con opción de salir de presentación que nos llevará otra vez al curso.



Figura 13

Aspecto de la plataforma con un curso (figura 14). Pulsando sobre “ver detalles” se nos mostrará el curso y pulsando sobre “la papelera” podremos eliminar el curso y todo su contenido.

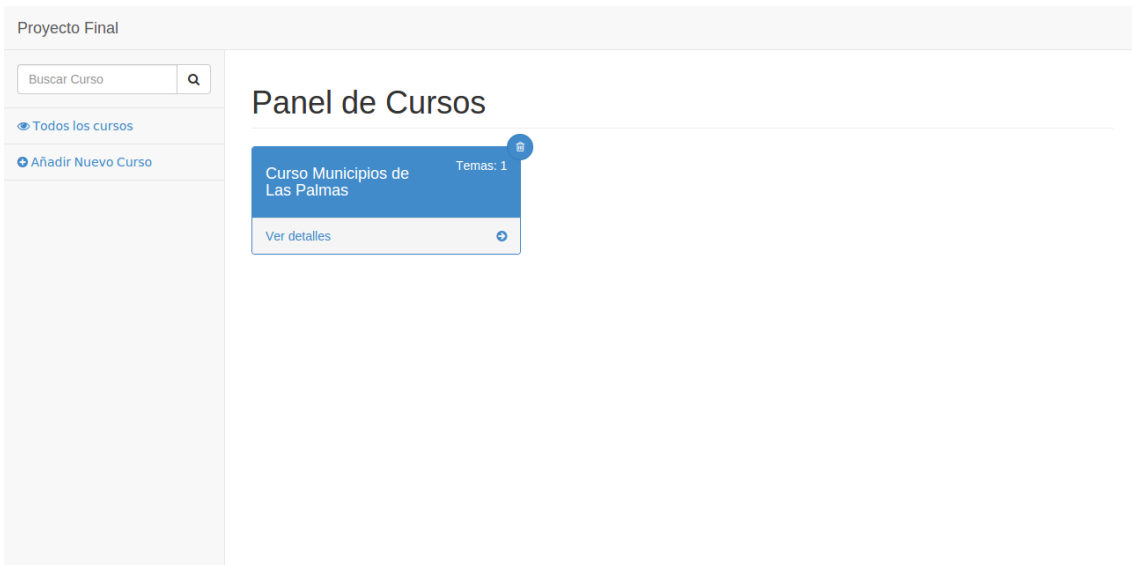


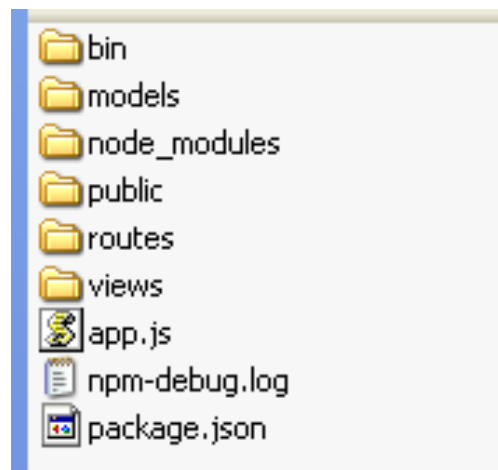
Figura 14

5.3 Desarrollo de INSUDOC

5.3.1. Estructura de la aplicación

La aplicación está desarrollada en Node.js, usando el framework Express.js. Este framework nos ofrece una estructura bien definida para comenzar con nuestra aplicación.

Express.js genera automáticamente la siguiente estructura:



- app.js. Este es el fichero que contendrá toda la configuración de nuestra aplicación; los módulos a usar, la configuración de la base de datos, la definición de las rutas, ...

Ejemplo fichero app.js

```
// Módulos requeridos por nuestra aplicación
var express = require('express');
var path = require('path');
var favicon = require('serve-favicon');
var logger = require('morgan');
var cookieParser = require('cookie-parser');
var bodyParser = require('body-parser');
var multipart = require('connect-multiparty');
var mongoose = require('mongoose');

var routes = require('./routes/index');
var cursos = require('./routes/curso');

var app = express();

// view engine setup
app.set('views', path.join(__dirname, 'views'));
app.set('view engine', 'jade');

app.use(logger('dev'));
app.use(bodyParser.json());
app.use(bodyParser.urlencoded({ extended: false }));
```

```
app.use(cookieParser());
app.use(express.static(path.join(__dirname, 'public')));

app.use(multipart());

// Definición de las rutas de la aplicación
app.use('/', routes);
app.use('/curso', cursos);
```

- **models.** En esta carpeta generaremos los diferentes modelos de datos que se usarán en nuestra aplicación. Estos modelos nos permitirán crear objetos del tipo “Cursos” que posteriormente se podrá almacenar en la base de datos.

Ejemplo del objeto Curso

```
var Curso = new Schema({
  titulo: { type: String, required: true },
  numeroTemas: { type: Number, required: true },
  temas: [Temas],
  creado: { type: Date, default: Date.now }
});
```

- **node_modules.** En esta carpeta se almacenará todos los módulos de terceros usados en nuestra aplicación (express, mongoose, mongodb, ...)
- **public.** En esta carpeta se almacenan todos los ficheros públicos de la aplicación, tales como imágenes, ficheros css y ficheros javascript usados en el cliente.

Ejemplo del fichero index.js:

```
$(function() {
  var idCurso;
  $(".borrar-curso").mouseenter(function() {
    $(this).removeClass("btn-primary").addClass("btn-danger");
  });

  $(".borrar-curso").mouseleave(function() {
    $(this).removeClass("btn-danger").addClass("btn-primary");
  });

  $(".borrar-curso").click(function() {
    $("#modalEliminarCurso").modal("show");
    idCurso = $(this).data("id-curso");
  });

  $("#boton-eliminar-curso").click(function() {
    $.ajax({
      type: "POST",
      url: "/curso/eliminar/",
      data: { idCurso: idCurso },
      dataType: "json",
    }).success(function(response) {
      console.log("ok");
      $("#modalEliminarCurso").modal("hide");
      if(response.status == 204) {
        location.reload();
      } else if(response.status == 404) {
        $("#modalAvisoCurso .texto").html("No se ha encontrado el curso");
        $("#modalAvisoCurso").modal("show");
      }
    });
  });
});
```

```

    }else {
      $("#modalAvisoCurso .texto").html("Se ha producido un
error. No se ha podido eliminar el curso");
      $("#modalAvisoCurso").modal("show");
    }
  }).error(function() {
    console.log("fail");
  });
});

$("#boton-buscar").click(function() {
  var data = $("#buscar-texto").val();
  window.location="/buscar/" + data;
});
});

```

- routes. En esta carpeta se crearan los ficheros que definen las rutas de nuestra aplicación así como los controladores encargados de obtener y generar datos. Ejemplo de ruta y su correspondiente controlador:

```

/* Página de inicio. Obtiene todos los cursos que hay en la base
de datos y genera una respuesta renderizando la vista
correspondiente */
router.get('/', function(req, res) {
  Cursos.find(function(err, cursos) {
    if(!err) {
      res.render('index', {cursos: cursos});
    } else {
      res.render('index', {error: "Se ha producido un error"});
    }
  });
});

```

- views. En esta carpeta se almacenarán las vistas de nuestra aplicación. Ejemplo de vista:

```

extends layout

block content
  .row
    .col-lg-12
      h1(class='page-header') Panel de Cursos
  .row
    if cursos.length == 0
      .alert.alert-warning.text-center No se han encontrado
cursos
    else
      each curso in cursos
        .col-lg-4.col-md-6
          .panel.panel-primary
            .panel-heading
              .row
                .col-xs-12.text-right
                  button(type="button", data-id-
curso="#{curso._id}", class="btn btn-primary btn-circle borrar-
curso", style="position: absolute; margin-top: -25px;")
                    i.fa.fa-trash-o
              .row

```

```
.col-xs-8
  h4 #{curso.titulo}
.col-xs-4.text-right
  .huge Temas: #{curso.numeroTemas}

a(href='/curso/id/#{curso._id}')
  .panel-footer
    span.pull-left Ver detalles
    span.pull-right
      i.fa.fa-arrow-circle-right
  .clearfix
```

5.3.2. Desarrollo

Para comenzar el desarrollo de esta aplicación tuve que entender la estructura de un documento .docx generado en LibreOffice y guardado como xml.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<?mso-application progid="Word.Document"?>
<w:wordDocument
xmlns:w="http://schemas.microsoft.com/office/word/2003/wordml"
xmlns:wx="http://schemas.microsoft.com/office/word/2003/auxHint"
xmlns:o="urn:schemas-microsoft-com:office:office"
xmlns:aml="http://schemas.microsoft.com/aml/2001/core"
xmlns:dt="uuid:C2F41010-65B3-11d1-A29F-00AA00C14882"
xmlns:v="urn:schemas-microsoft-com:vml" xmlns:w10="urn:schemas-
microsoft-com:office:word"
xmlns:number="urn:oasis:names:tc:opendocument:xmlns:datastyle:1.0"
xml:space="preserve" w:embeddedObjPresent="no"><o:DocumentProperties
xmlns:fo="urn:oasis:names:tc:opendocument:xmlns:xsl-fo-
compatible:1.0"><o:Title/><o:Subject/><o:Keywords/><o:Description/><o:
Category/><o:Author/><o:LastAuthor/><o:Manager/><o:Company/><o:Hyperli
nkBase/><o:Revision>2</o:Revision><o:TotalTime>0</o:TotalTime><o:LastP
rinted/><o:Created>2014-12-04T18:45:00Z</o:Created><o:LastSaved>2014-
12-
04T21:59:12.737595555Z</o:LastSaved><o:Pages>4</o:Pages><o:Words>127</
o:Words><o:Characters>680</o:Characters><o:Paragraphs>40
...
```

Una vez localizadas las partes del documento que me interesaban: etiquetas del esquema numerado, imágenes, etc. parseamos el fichero buscando estas etiquetas y almacenándolas en un array, dividiéndolas por tipos (imágenes por un lado y títulos por otro). Una vez terminado de recorrer el documento de texto y almacenado en el array los datos que buscamos, generamos un **objeto tipo curso** donde introduciremos los datos del vector y posteriormente guardamos este objeto en nuestra base de datos. Al ser una base de datos NoSQL el objeto curso se guarda como un Json.

En este momento ya tendríamos creado un curso, por lo que ahora podemos crear la correspondiente diapositiva. Para crear ésta, lo que hacemos es mostrar en una pantalla todos los títulos e imágenes que contenía el documento. Justo debajo de cada título, hay un botón “check” que al marcarlo establecemos lo que contendrá una diapositiva hasta ese punto. Dentro del **objeto curso** generaremos un **objeto tipo diapositiva** que contendrá los textos e imágenes que hemos definido con anterioridad.

5.4 Metodología de trabajo

En cuanto a mi metodología de trabajo usada, he intentado combinar metodología de desarrollo Ágil y el método que más se acerca a mi trabajo es la técnica Pomodoro, adaptando esta a mis necesidades

Técnica Pomodoro

La Técnica Pomodoro es un método para la administración del tiempo desarrollado por Francesco Cirillo a fines de los años 1980.¹ La técnica usa un reloj para dividir el tiempo dedicado a un trabajo en intervalos de 25 minutos -llamados 'pomodoros'- separados por pausas. Relacionado con conceptos como timeboxing y desarrollo iterativo e incremental, usados en el desarrollo de software, el método ha sido adoptado en contextos de desarrollo ágil y Programación en pareja(pair programming)² además de otros contextos de trabajo.

El método se basa en la idea de que las pausas frecuentes pueden mejorar la agilidad mental,^{3 4} y trata de ofrecer una respuesta eficaz frente al tiempo, en lugar del estado de ansiedad que suele provocar el "devenir" del tiempo, del que se habla en los escritos de Henri Bergson y Eugene Minkowski.

Son fundamentales las etapas de **planeamiento, anotación, registro, proceso y visualización.**

En el planeamiento, las tareas son priorizadas en una lista "Para hacer hoy". Esto permite a los usuarios estimar el esfuerzo que cada tarea puede requerir. Cuando se completan los "pomodoros" (en lo que se refiere a la técnica, 'pomodoro' significa un periodo indivisible de tiempo, usualmente de 25 minutos), estos se registran, dando un sentimiento de logro y, a la vez, datos base para una posterior auto-observación y mejora. Un objetivo esencial de la técnica es eliminar las interrupciones, tanto internas como externas. Esto se hace registrándolas y posponiéndolas siempre que sea posible.

Hay cinco pasos básicos para implementar la técnica pomodoro:

1. Decidir la tarea a realizar
2. Poner el pomodoro (el reloj o cronómetro) a 25 minutos
3. Trabajar en la tarea hasta que el reloj suene y anotar una X
4. Tomar un pausa breve (5 minutos)
5. cada cuatro "pomodoros" tomar una pausa más larga (15-20 minutos)

Mi adaptación se ha basado sobre todo aumentando el tiempo, ya que los pomodoros de 25 minutos se me quedaban algo escaso cuando estaba programando, lo que más me ha gustado es el marcarme las tareas a realizar y cuantificarlas en prioridad/tiempo, realizando las tareas que menos tiempo me requería en primer lugar.

Capítulo 6

6.1 Conclusiones

El trabajo fin de grado tiene por objetivo la elaboración de un proyecto informático aplicando los conocimientos adquiridos a lo largo de las asignaturas del Grado en informática.

Como se ha visto a lo largo de esta memoria, los objetivos iniciales han sido superados, requiriendo de una serie de etapas para realizar dicho propósito.

Centrándome ahora en **INSUDOC**, se pretende que llegue a ser una potente herramienta que facilite la labor del docente, entre otros.

La funcionalidad prevista se ha aumentado con el fin de aprovechar el software libre y la potencia que las nuevas tecnologías TIC nos permiten hoy en día, ya que la idea del proyecto no iba encaminada hacia las herramientas Web.

El tiempo transcurrido desde la elección del trabajo hasta la finalización del mismo, ha sido de 12 meses.

Cabe destacar los conocimientos técnicos adquiridos en el desarrollo de este trabajo, ya que para alcanzar los diferentes objetivos, ha sido necesario dominar muchas áreas.

En definitiva, el desarrollo de este trabajo de fin de grado, ha significado un antes y un después a nivel personal. He aprendido y comprendido todos los conceptos y problemas que rodean la planificación, puesta en marcha, documentación y elaboración de un proyecto de esta magnitud tecnológica, con la satisfacción de haber alcanzado el objetivo final por muy lejos que se viese en sus comienzos. No hay nada más gratificante que trabajar en algo que a uno le guste y en lo que crea.

Como Conclusión económica y línea de negocio es un proyecto:

- Ambicioso porque entra en un sector con gran presente y futuro , pudiendo alcanzar el liderazgo de este sector (con este producto específico) en un período aproximado de corto a medio plazo,
- Innovador ya que no existe una aplicación parecida, y si existe no es muy conocida
- Realizable como modelo de negocio porque tenemos un cliente potencial interesado en ella y además el desarrollo será hecho en gran parte por mi

6.2 TRABAJO FUTURO

Como opciones de futuro, veo la posibilidad de integrar esta herramienta directamente en los software (documentos de textos) ya sean libres o no, ya que hasta ahora solamente se ha probado con el libreoffice, y por qué no, también realizar una presentación de una hoja de cálculo o Base de datos

Objetivos:

A corto/medio plazo

- 1) Ir dotando de mayor funcionalidad al producto para optimizar sus prestaciones.

Evolución a largo plazo:

- 2) Lanzamiento Mundial: si el producto supera las estimaciones realizadas inicialmente y se convierte en un éxito en España, se lanzaría a nivel mundial

6.3 Fuentes de información

- Tutoriales Onlines Web

www.3schools.com

- Node.Js

www.nodejs.org

- jquery

www.jquery.com

- mongo.db

www.mongodb.org

- JavaScript

<https://developer.mozilla.org/es/docs/Web/JavaScript>

- Tutoriales/ fuente de información

<https://es.wikipedia.org/>

www.getbootstrap.com

- libreoffice

<https://es.libreoffice.org/>

- Lenguajes de marcas

es.wikipedia.org/wiki/Extensible_Markup_Languag

es.wikipedia.org/wiki/Lenguaje_de_marcado