



INGENIERÍA DE TELECOMUNICACIÓN

Curso Académico 2012/2013

Proyecto Fin de Carrera

DISEÑO Y DESARROLLO DE UNA APLICACIÓN
WEB PARA LA CONCESIÓN DE TRABAJOS DE
FIN DE GRADO Y PROYECTOS DE FIN DE
CARRERA PARA LA ETSIT-URJC
INCORPORANDO ELEMENTOS DE HTML5 Y
CSS3

Autor : Laura Martínez Borlaff

Tutor : Gregorio Robles

Resumen

En este documento explicamos todos los puntos del desarrollo de este proyecto de fin de carrera que trata de una aplicación web para la concesión de trabajos de fin de grado (T.F.G.). Facilitamos un sitio de encuentro entre profesores y alumnos para que éstos se elijan entre sí para trabajar juntos de una manera oficial dentro de la Universidad, más concretamente la ETSIT, y pública para todos los alumnos, gozando de las mismas oportunidades.

En esta herramienta los alumnos podrán ver todas las ofertas vigentes y solicitar las que más les interesen. Los profesores podrán publicar sus ofertas de trabajos y les ayudamos en la tarea de deliberación del candidato que realice un oferta, dándole acceso a datos concluyentes para esta decisión.

Para el desarrollo de este proyecto hemos utilizado recursos de HTML5 y CSS3 para crear una interfaz más atractiva, en la cuál, alumnos y profesores se sientan a gusto a la hora de usar. De igual forma, hemos tratado que su uso sea de lo más intuitivo y sencillo para que un usuario pueda disfrutar de todas las posibilidades que ofrecemos desde el primer instante en que se crea la cuenta y no necesite de ningún procedimiento adicional.

Índice general

1. Introducción	1
1.1. Prólogo	1
1.1.1. Breve historia sobre las webs	1
1.1.2. Repaso sobre la historia de las redes sociales	3
1.1.3. Tecnologías relacionadas	4
1.2. Motivación	9
1.3. Estructura de esta memoria	10
2. Objetivos	11
2.1. Creación de una aplicación web	11
2.2. Creación de un perfil de usuario	11
2.3. Creación de un formato esquemático para las ofertas	12
2.4. Diseño de la lógica de asignación y aceptación de las ofertas	12
2.5. Interfaz atractiva	13
2.6. Estudio de soluciones	13
2.7. Planificación inicial	13
3. Estudio de soluciones	15
3.1. Moodle	15
3.1.1. Características de Moodle	16
3.1.2. Soluciones estudiadas	18
3.2. Mahara	24
3.2.1. Integración con Moodle	24
3.3. Conclusión sobre las diferentes soluciones	29
4. Desarrollo de la herramienta para la concesión de T.F.G.	33
4.1. Introducción	33
4.2. Especificaciones técnicas	34
4.2.1. ¿Por qué PHP?	34
4.2.2. HTML5 Y CSS3	37

4.3.	Desarrollo de la herramienta web	37
4.3.1.	Descripción de la herramienta	37
4.3.2.	Modelo Vista Controlador (M.V.C.)	41
4.3.3.	Explicación de la herramienta	42
4.3.4.	Problemas encontrado durante el desarrollo de la herramienta	84
4.4.	Resumen	85
5.	Conclusiones	87
5.1.	¿Qué cosas de las aprendidas durante la carrera he aplicado?	88
5.2.	¿Qué cosas nuevas he aprendido?	88
5.3.	Problemas en la planificación	89
5.4.	Líneas futuras	89
A.	Manual de usuario para THAIS	91
A.1.	Usuario alumno	91
A.1.1.	Registro y creación de cuenta de usuario en THAIS	91
A.1.2.	Inicio de sesión	94
A.1.3.	Página de inicio	94
A.1.4.	Página de una oferta	95
A.1.5.	Página de TFG concedidos	96
A.1.6.	Perfil de usuario	97
A.1.7.	Editar perfil de usuario	97
A.1.8.	Notificaciones de usuario	98
A.1.9.	Cambiar contraseña	99
A.1.10.	Cerrar sesión	99
A.2.	Usuario profesor	99
A.2.1.	Página de inicio	99
A.2.2.	Página de TFG concedidos	100
A.2.3.	Página para crear una nueva oferta	100
A.2.4.	Página mis ofertas	101
A.2.5.	Página de una oferta	102
A.2.6.	Página de notificaciones	103
A.2.7.	Cambiar contraseña	104
A.3.	Usuario administrador	104
A.3.1.	Preparación del entorno	104
A.3.2.	Registro y creación de cuenta	105
A.3.3.	Página de inicio	105
A.3.4.	Página TFG concedidos	106

A.3.5. Página de una oferta	107
A.3.6. Página para crear una nueva oferta	107
A.3.7. Página con listado de usuarios	107
A.3.8. Página de cambio de rol a usuarios	108
A.3.9. Página de cambio de contraseña a usuarios	108
A.3.10. Página de cambio de contraseña	109
A.3.11. Página para eliminar usuarios	109
B. Instalación de Moodle y tecnologías requeridas	111
B.1. Preparación del entorno	111
B.1.1. Comprobando que el entorno es el adecuado	112
B.2. Configuración de la base de datos para Moodle	112
B.2.1. Creación de la carpeta para los datos de Moodle	113
B.3. El paquete Moodle	113
B.4. Problemas encontrados	115
B.4.1. MySQL-SERVER	115
B.4.2. PHPMYADMIN	116
Bibliografía	117

Capítulo 1

Introducción

1.1. Prólogo

Desde que surgió el fenómeno de las páginas *web* hasta hoy en día, éstas han experimentado una notable transformación, tanto en la forma en que son desarrolladas como en la forma en que los usuarios hacen uso de ellas. Por otro lado, los lenguajes de programación empleados para desarrollar páginas web han ido evolucionando, permitiendo la creación de páginas cada vez más ligeras, activas y atractivas. Los usuarios han ido notando que ahora van teniendo más y más control sobre ellas.

Desde sus comienzos, podemos clasificar los distintos tipos de páginas web que han ido surgiendo según distintos criterios: según su audiencia (pública, extranet o intranet), según su grado de dinamismo (sitios interactivos o sitios estáticos), su apertura (estructura abierta, estructura cerrada o estructura semicerrada), por su profundidad, por sus objetivos (comerciales, informativos, ocio, navegación, artísticos o personales) [2]; ya que no hay una manera general y unificada para definir cada una de sus clases [1].

1.1.1. Breve historia sobre las webs

Centrándonos en las distintas versiones según el *World Wide Web* (WWW), [3], o también conocido como *Red Informática Mundial*; la *Web 1.0* surgió en 1991 y consistía en un diseño lineal, que podía incluir texto, gráficos, etc. Éstos eran almacenados en servidores de *hosting* en HTML, la finalidad era sólo de lectura, y su característica principal era la ausencia de interacción con el usuario. Esta característica conformaba lo que conocemos como web estática; que en otras palabras quiere decir que el usuario seguía un modelo unidireccional, sólo leía la información que mostraban dichas páginas, era un mero espectador, por así decirlo, como se puede observar en la figura 1.1 [17].

Más adelante, con la aparición de la *Web 2.0* en el año 2004, fuimos testigos del paso de esa unidireccionalidad hasta una bidireccionalidad entre usuario y web. El usuario podía transmitir



Figura 1.1: Esquema del funcionamiento de la Web 1.0. Fuente [17]



Figura 1.2: Esquema del funcionamiento de la Web 2.0. Fuente [17]

información a la web, por ejemplo, por medio de comentarios o respuestas, y recibir información de la web basándose en el sistema que aportó su versión previa (véase figura 1.2) [4]. Esta versión 2.0 se define de forma dinámica, es decir, cuando el usuario visita el sitio web y tras las interacciones realizadas por éste, la página se construye y además no necesita actualizarse mediante un servidor. Esto creó la necesidad de una infraestructura para almacenar tanto las acciones realizadas por el usuario como la información y de ahí el uso de las bases de datos. Esta versión es la que sigue vigente actualmente, pero durante los últimos años y, gracias a la tenacidad de Tim Berners-Lee¹, ha tendido hacia una web semántica. Esto ha mejorado la forma de buscar información en Internet, haciéndolo más rápido y obteniendo resultados de búsqueda más personalizados, como se refleja en la figura 1.3 ([4], [17], [5] y [6]). Uno de los factores que han hecho esto posible ha sido el enorme intercambio de información que han promovido las redes sociales.

¹Timothy "Tim" John Berners-Lee es considerado como el padre de la web y es el director de *World Wide Web Consortium (W3C)*. [22]

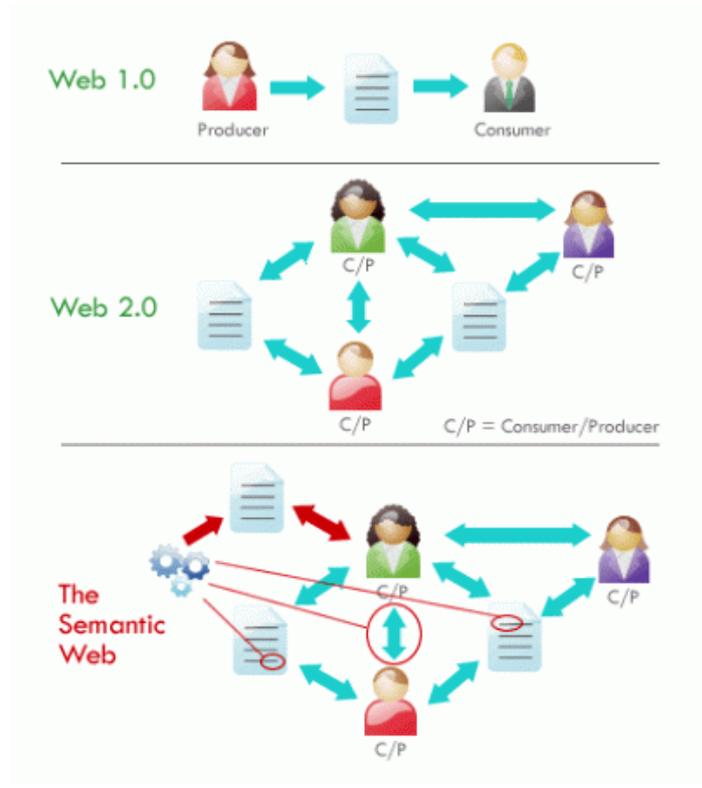


Figura 1.3: Esquema del funcionamiento de la Web semántica. Fuente [18]

1.1.2. Repaso sobre la historia de las redes sociales

Por su parte, las redes sociales, sin lugar a dudas, han marcado un antes y un después en la manera de navegar por Internet, de intercambiar información de forma fácil y sencilla, conectando personas alrededor de todo el mundo. Las redes sociales, actualmente, gozan de una increíble cobertura mundial, donde personas cotidianas, y no sólo expertos, muestran su conocimiento o comparten gustos de una forma ágil, rápida y de bajo coste con el resto de personas, a veces, haciendo de enlace entre personas de otros continentes, lo que lleva consigo un gran potencial económico [7].

En 1994 empezaron a ver la luz portales que llevaban las primeras pinceladas de lo que conocemos hoy en día por red social. Eran portales que permitían a los usuarios expresar sus comentarios en foros y servicios de mensajería instantánea; se trataba de redes muy localizadas, pero sin mucha relevancia a nivel mundial, y con acceso restringido, debido a que no fue hasta 1995 cuando se empezó a extenderse el uso de Internet. Fue en 1997 cuando apareció *Sixdegrees.com*, conocida como la primera red social, que permitía a sus usuarios tener un perfil, para poder definirse, y una lista de amigos. En el año 2000 fue cuando se produjo el *boom* de Internet, llegando a todas partes del mundo. Dos años más tarde de esa explosión, en 2002, surge *Friendster*, la primera red que daba la posibilidad de conectar de manera *online* a personas

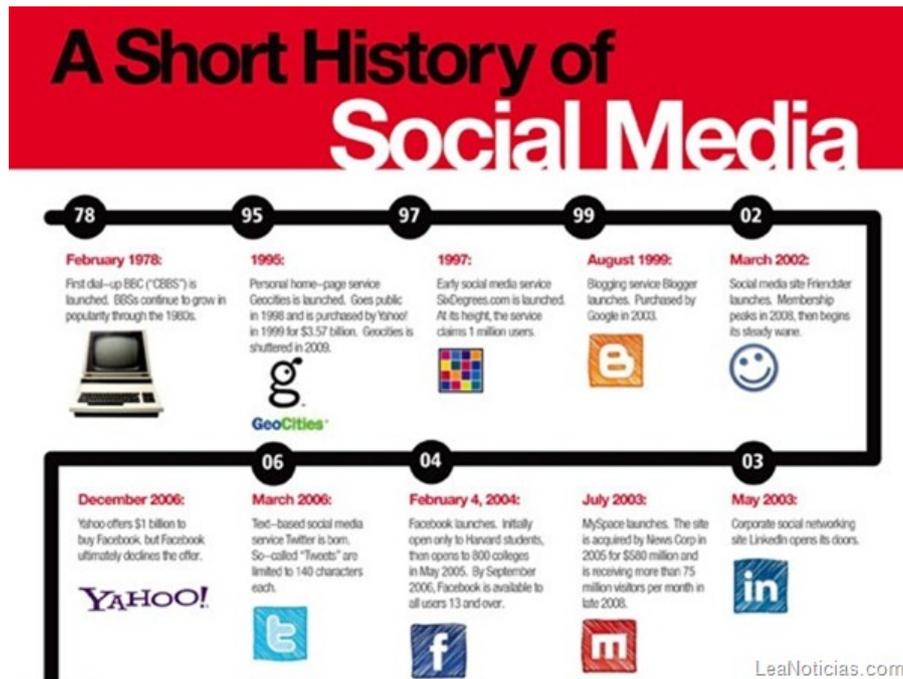


Figura 1.4: Línea de tiempo/resumen de las redes sociales más importantes. Fuente [19]

y consiguió en un mes llegar a los tres millones de usuarios. Sin embargo, fue igual de grande su éxito como su fracaso, ya que no fue capaz de manejar, en lo que se refiere técnicamente a la gran cantidad de usuarios que demandaban su portal. En 2003, Tom Anderson sacó a la luz *MySpace*, aprovechándose de la caída de *Friendster*, pero después, en 2004 con la aparición de *Facebook*, *MySpace* quedó obsoleta. *Facebook* consolidó y masificó las redes sociales. Inicialmente fue creada para los estudiantes de la Universidad de Harvard, pero su extensión fue tal que cuenta con 900 millones de usuarios a fecha de abril de 2012 [23] (Figuras 1.4 y 1.5). Muchos países cuentan, además, con otras redes sociales creadas en el propio país y principalmente dirigida a los usuarios de dicho país, como por ejemplo: *Tencent QQ* en China², *Mixi* en Japón³, *StudiVZ* en Alemania⁴ o *Tuenti* en España⁵ [8] [9] [10].

1.1.3. Tecnologías relacionadas

Debido a la gran acogida que han tenido las redes sociales, era obvio que tuviera una repercusión en el terreno educativo. Hoy en día se está haciendo uso de comunidades de usuarios para la comunicación entre alumno y profesor, ya que estamos en un momento donde las personas pasan la mayor parte de su tiempo conectados vía Internet (véase figura 1.6). Las redes sociales han hecho una distinción de los usuarios en dos grandes grupos: *digital immigrants* y

²Website: <http://www.qq.com>. Internacional site: <http://www.imqq.com>

³Website: <http://mixi.jp>

⁴Website: <http://www.studivz.net>

⁵<http://www.tuenti.com>

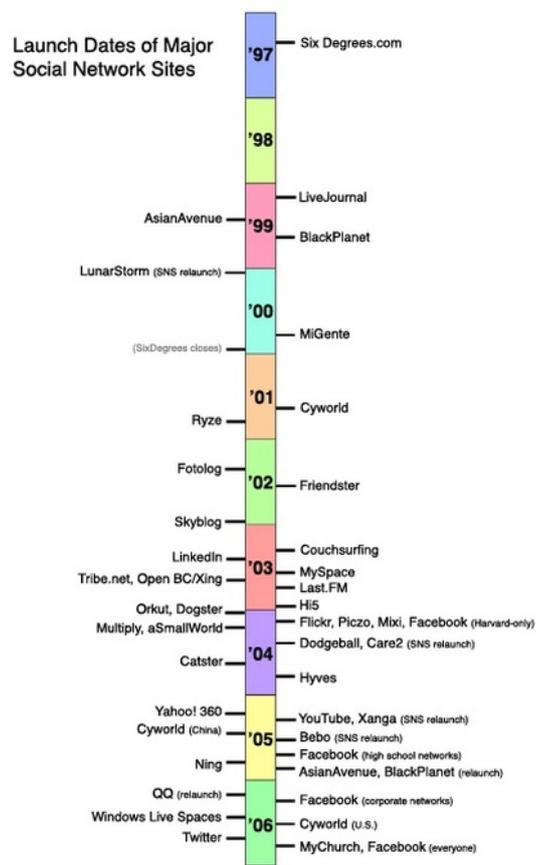


Figura 1.5: Línea de tiempos de las redes sociales. Fuente [20]



Figura 1.6: Línea de tiempos de las redes sociales. Fuente [21]

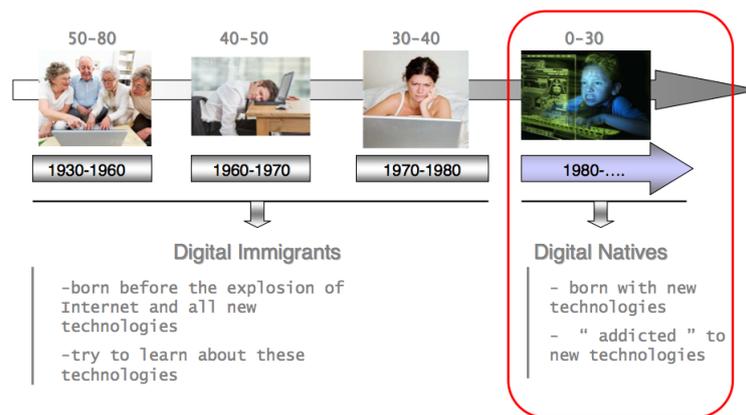


Figura 1.7: Línea de tiempos de las redes sociales. Fuente [21]

digital natives según se muestra en la figura 1.7. Mientras los primeros, dentro del campo de la educación, serían representados por los profesores y son los que tratan de aprender sobre las nuevas tecnologías que van apareciendo. Los alumnos, *digital natives*, serían quienes han nacido con ellas y están totalmente familiarizados [21]. Los nacidos en la generación de los *digital natives* no asimilan los conocimientos y estudian de la misma manera que los anteriores, aprenden más por imágenes que por palabras, y dan mucha más importancia al debate de ideas y opiniones, algo bastante coherente, ya que dentro de esta facilidad de difusión de información es primordial el acceso a datos "frescos" para cualquier estudiante, sobre todo para estudiantes de ingeniería como nosotros.

Una de las herramientas más usadas en la educación es el *Moodle*, perteneciendo al conjunto de aplicaciones web conocidas bajo el nombre de *LMS (Learning Management System)*. Es

una herramienta enfocada a facilitar la labor de la docencia y diseñada teniendo en cuenta cómo funciona la mente de un alumno. Su creador, un ingeniero informático llamado Martin Dougiamas, no estaba conforme con las aplicaciones web del momento para el apoyo educativo. Martin realizó la carrera pedagógica para conocer las claves del constructivismo y así construir una herramienta en base a ello. Es una aplicación de fácil uso para el alumno y el profesor, donde se puede seguir de forma sencilla la evolución de cada alumno, mejora la comunicación de dudas generales o los conceptos que han quedado poco claros al alumnado, detectando y previniendo los atrasos de éstos [13] [15] [16].

Pero no sólo ha llegado el *boom* de Internet y las redes sociales a las aulas, también ha conquistado el terreno empresarial. Las empresas se han ido adaptando según avanzaba la tecnología, en concreto y en lo que concierne a este proyecto, en la manera de reclutar personal. Antes se utilizaban medios clásicos, pero desde el auge de las redes sociales, muchas empresas se han sumado a esta corriente y buscan candidatos por redes orientadas a los negocios. De entre las que destaca de forma internacional *LinkedIn*. LinkedIn fue fundada por Reid Hoffman en 2002, pero no comenzó a funcionar hasta 2003, fecha anterior a Facebook [11] [12]. Se trata de una red profesional que favorece los contactos de negocio empresarial. En ella, cada usuario cuenta con un perfil en el que muestra sus conocimientos, estudios, gustos e intereses, en un formato parecido al de un currículum vitae, esforzándose para mantenerlo lo más actualizado posible. Tras este paso, los usuarios pueden ponerse en contacto por igual con compañeros de estudio, trabajo, profesores, etc. Las empresas que emplean LinkedIn, disminuyen riesgos en la contratación, ya que pueden seleccionar aquellas personas *conocidas* para la búsqueda de candidatos que reúnan las especificaciones requeridas para la oferta. Muchas empresas tienen la filosofía de escoger candidatos recomendados por sus actuales empleados, ya que eso facilita en gran medida la labor del proceso de selección. Los candidatos que son recomendados tienen más probabilidades hacerse con un puesto de trabajo frente a los que no, ya que implica saber a priori el nivel de conocimiento que tiene el individuo para el trabajo requerido, la actitud que toma en su trabajo, su anterior experiencia y la trayectoria profesional que ha tomado y quiere tomar, además de aptitudes como el trabajo en equipo, liderazgo o la capacidad de desenvolverse bajo presión.

En el curso 2009-2010, la ETSI de Telecomunicación de la URJC inauguró un portal web de la Escuela con el soporte Moodle. Esta iniciativa pretendía mejorar la comunicación entre la Escuela como tal y sus alumnos de forma que representara un lugar de encuentro de información entre todos los integrantes de la ETSIT-URJC.

La versión actual de Moodle de la ETSIT-URJC es la 2.0, pero para llegar hasta ella, Moodle ha ido incorporando diferentes servicios durante todas sus versiones, algunos de ellos bastante significativos. La versión 1.0 permitía diferentes perfiles de usuario, distinguiendo entre alumno, profesor y administrador, utilizaba herramientas de comunicación como Forum, Sur-



Figura 1.8: Top 10 de países con instalaciones Moodle. Fuente [14]

vey y journal, y disponía de una página principal y distintos espacios. En la versión 1.1, con año de aparición en el 2003, se incorporaron al núcleo las funciones de *Workshop* y chat, además en la versión 1.2 se añadieron otras funciones como poder juntar a los alumnos en grupo dentro de un mismo curso y la función *Filter* que aceptaba texto multi-idioma y censor de palabras. En la versión 1.4 se caracterizó por la automatización de la base de datos, la función *WIKI* y la completa modularización de los módulos. La versión 1.8 destacó por su accesibilidad, servicios web y, sobre todo, por establecer una red Moodle donde las distintas plataformas podían comunicarse. Finalmente, con la versión 2.0 y la que actualmente hace uso la universidad, mejoró sus características orientadas a la comunicación y las necesidades de las comunidades de usuarios [15].

Moodle, sin lugar a dudas, supone una de las herramientas más usadas a nivel mundial como se puede ver en las figuras 1.8, 1.9 y 1.10, las cuales muestran la expansión global que ha tenido desde su creación.

Una vez comprobada la eficacia de este portal para la comunicación entre los miembros de la ETSIT es momento de extender sus servicios. En este sentido, dada la situación de crisis económica actual, parece sensato desarrollar servicios dirigidos hacia la mejora de la situación

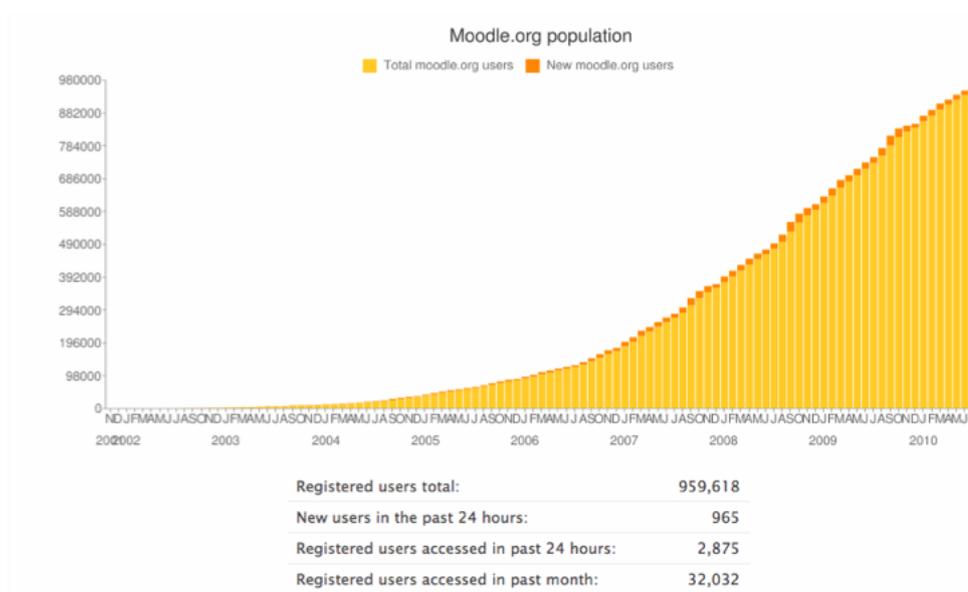


Figura 1.9: Gráfica sobre el número de usuarios registrados. Fuente [15]

Moodle Statistics

Registered sites	68,026
Countries	219
Courses	6,511,135
Users	60,902,294
Teachers	1,273,806
Enrolments	53,026,250
Forum posts	107,113,281
Resources	57,717,404
Quiz questions	122,437,482

Figura 1.10: Estadísticas sobre el uso de Moodle. Fuente [14]

laboral de los alumnos ya egresados o en los últimos cursos de carrera de la ETSIT-URJC.

1.2. Motivación

En este proyecto se han querido combinar las funcionalidades de la Web 2.0 con la idea en la que se mueve LinkedIn y para crear una aplicación web para la asignación de los trabajos de fin de grado. La finalidad que persigue es ayudar a la comunicación entre los alumnos y profesores de la Escuela Técnica Superior de Ingenieros de Telecomunicación, a la hora de elegir un proyecto de fin de carrera o trabajo de fin de grado.

Existe una herramienta actual propia de la Universidad Rey Juan Carlos, pero debido a los problemas que genera, hemos pensado en una herramienta propia de la Escuela para este mismo

proceso. Creando una red interna de ofertas para el beneficio de los alumnos de la ETSIT-URJC y fomentando una comunicación más fluida entre la Universidad, la Escuela, sus alumnos y los profesores. Las ofertas que se publiquen pueden estar dirigidas a alumnos de todos los cursos y que quieran involucrarse en un proyecto de la ETSIT. No sólo limitando el uso de la herramienta a alumnos de último curso.

A pesar de que la Universidad cuenta con una bolsa de ofertas de este tipo y da servicio a todos los alumnos de todos los distintos grados, pero no está centrada para los alumnos de Ingeniería en Telecomunicación, que sería lo más idóneo. Por todo ello, es habitual ver ofertas para este grupos alumnos en las que son los propios profesores quienes las publican vía Moodle con la finalidad de llegar a todos los alumnos de la Escuela, lo que resultaba muy desordenado puesto que estas ofertas se mezclaban con otro tipo de noticias que poco o nada tenían que ver sobre la Escuela.

Este proyecto se ha pensado como una solución a esta situación, creando una comunidad en la que todo alumno de Ingeniería en Telecomunicación formará parte y será notificado. Además podrá solicitar aquellas ofertas que le interesen sin necesidad de otros procedimientos.

1.3. Estructura de esta memoria

Esta memoria se divide en tres grandes capítulos: estudio de soliciones, desarrollo de la aplicación web y conclusiones. En el primero de ellos explicamos todas las posibles soluciones que hemos investigado para hacer llegar la herramienta a nuestro público. En el segundo capítulo entramos en detalle sobre cómo hemos implementado la plataforma y, también, explicamos su funcionamiento. En el capítulo de conclusiones hacemos un breve resumen por todos puntos que hemos estudiado en este proyecto y añadimos ideas para el futuro desarrollo de nuestra aplicación.

Además, en la parte final de este documento hemos agregado unos anexos. Uno de ellos es un manual de usuario de nuestra herramienta, donde se encuentran explicadas todas las funciones que puede realizar cualquier tipo de usuario en ella. Y el otro es una guía de instalación de Moodle.

Capítulo 2

Objetivos

El objetivo principal de este proyecto fin de carrera será el de crear un servicio orientado hacia la mejora de la búsqueda y concesión de trabajos de fin de grado para los alumnos de la ETSIT-URJC. Este objetivo principal se particulariza en los siguientes objetivos específicos de cada apartado.

2.1. Creación de una aplicación web

Creación de una aplicación web para alumnos y profesores de la Escuela orientada para simplificar la adjudicación de trabajos de fin de grado. En esta nueva herramienta, y siguiendo el esquema que tanto éxito ha tenido LinkedIn, todos los alumno contarán con un perfil que seguirá el esquema de un curriculum vitae, aunque con unas particularidades distintas. El primer paso será definir las acciones que debe contener nuestra herramienta y diseñar un primer borrador de ella.

Como primer requisito, nuestra herramienta deberá constar de una página principal donde los usuarios pudieran ver todas las ofertas para estos trabajos que se han publicado. Una página con un historial de trabajos de fin de grado ya concedidos y un perfil para ellos usuarios.

2.2. Creación de un perfil de usuario

La idea a seguir es que cada usuario tenga un perfil, distinguiendo tres tipos de usuarios: alumnos, profesores y administrador. Donde proporcionarán datos sobre su educación, centrándonos más en campos relacionados con sus intereses dentro de la variedad de los campos que trata su grado, todos los conocimientos que crea importante resaltar, tanto si los ha adquirido por medio de esta Universidad como si los ha aprendido por otra fuente. Se ampliarán los campos a redactar por el usuario, con los típicos apartados en el perfil de cualquier herramienta como por ejemplo nombre, apellidos y correo de contacto. Más adelante se verá las modificaciones de esta idea inicial a lo que realmente luego en la práctica tuvimos que decidir

y por qué llegamos a esa conclusión. Como por ejemplo, que sólo los alumnos tuvieran perfil y no los profesores.

Con esta herramienta, se pretende que todos los alumnos accedan a las ofertas de una forma clara y más enfocada a un público dentro de la ETSIT, a diferencia de la plataforma actual orientada a toda la Universidad. Además, se dotará a nuestra aplicación de una función donde los alumnos puedan filtrar las ofertas según el campo dentro de la ingeniería de telecomunicación que más les interese.

2.3. Creación de un formato esquemático para las ofertas

Otro de los objetivos a cumplir es la de crear un formato de oferta en dicha red social para definir los posibles trabajos destinados a los alumnos. El formato a diseñar será siguiendo la línea de los esquemas de ofertas que se van publicando en el Moodle, pero aportando más campos que creemos que dan una mejor idea sobre el trabajo a realizar. Para que tanto alumnos como profesores se escojan con una menor probabilidad de error. Lo que se quiere evitar son los casos en los que un alumno se compromete con una oferta que realmente luego, cuando se mete de lleno, no le guste. O, viceversa, que un profesor asigne a un alumno para un trabajo suyo que luego no tenga el rendimiento que esperaba por falta de conocimientos previos necesarios.

En esta plantilla de ofertas los profesores podrán redactar una breve descripción sobre el trabajo, sus objetivos o tecnologías necesarias por ejemplo. Además deberá contener información de contacto con el responsable del proyecto.

2.4. Diseño de la lógica de asignación y aceptación de las ofertas

Nuestra herramienta necesita un protocolo con el que proceder en la asignación de las ofertas por parte de los profesores y su aceptación por parte del alumno. Para ello, la forma que pensamos en un principio era simplemente que los alumnos solicitaran las ofertas que más les gustara y los profesores pudieran deliberar qué candidato sería mejor para llevar a cabo el proyecto. Una vez decidido, el profesor otorgaría a ese alumno la oferta y se le notificaría de la manera oportuna.

Durante la realización del proyecto se verá que esta idea inicial tenía fallos, y se diseñó un protocolo de doble confirmación, es decir, los profesores asignan a un alumno un trabajo y éste decide si aceptar o no la oferta, puesto que pueden aparecer casos donde a un mismo alumno se le ofrezcan dos trabajos. Obviamente el alumno sólo puede realizar un trabajo de fin de grado y deberá elegir uno de ellos. Con este nuevo mecanismo se corrigió los posibles casos de error.

2.5. Interfaz atractiva

Nuestra intención es que esta herramienta sea del gusto de profesores y alumnos. Queremos que ambos se sientan cómodos con nuestra aplicación. Por eso, uno de nuestros objetivos es la creación de una interfaz amigable para ambos. Sin resultar muy informal para los profesores pero tampoco muy formal para los alumnos, ya que ambos serán nuestros clientes y tenemos que llegar a los dos por igual. Además, una de nuestras bases es hacer llegar la información de una forma clara y fácil para los dos tipos de usuarios. Teniendo en cuenta el uso que esperan de esta red cada uno de ellos, puesto que los profesores no harán el mismo uso que los alumnos. Para los profesores queremos facilitarles el procedimiento de publicar sus ofertas de proyectos y la tarea de elegir al mejor candidato. En cambio, queremos que los alumnos estén informados en todo momento de las distintas fases de este proceso de selección y que las ofertas de los profesores lleguen a la mayor parte de los alumnos, si no la totalidad. Para conseguir todos estos objetivos, debemos diseñar una interfaz con un estilo que sea del agrado de todos los usuarios.

2.6. Estudio de soluciones

En el desarrollo de esta aplicación se estudiarán las diferentes soluciones sobre cómo hacer llegar la herramienta a nuestro público deseado. Como primera opción se estudiará Moodle, puesto que es la plataforma más usada por alumnos y profesores. Buscaremos plugins o herramientas externas que en contacto con Moodle puedan ayudar a crear esta nueva aplicación. Más adelante se explicará cada una de los bloques que estudiamos y la decisión tomada a seguir.

2.7. Planificación inicial

La planificación inicial pensada para la realización de este proyecto fin de carrera se dividió según los distintos capítulos definidos en el índice, y, a cada uno de ellos se le estimó una duración como se muestra a continuación:

- **Agosto 2012:** Redacción de los capítulos primero, Estado del arte, y segundo, Objetivos.
- **Septiembre 2012:** Inicio del desarrollo del proyecto en Moodle: estudio de su arquitectura interna, investigación entre los *plug-in* de Moodle e implementación de un perfil de usuario ampliado.
- **Octubre 2012:** Familiarización con PHP, estudio para desarrollar una herramienta como un plugin de Moodle.

- **Noviembre 2012:** Estudio y familiarización de la herramienta Mahara para crear nuestra herramienta como un plugin en este ePortfolio.
- **Diciembre 2012:** Definición detallada de las funcionalidades básicas de la aplicación web. Inicio del desarrollo de la herramienta.
- **Enero 2013:** Desarrollo de la herramienta: página de inicio, perfiles de usuario y página de ofertas.
- **Febrero 2013:** Primer prototipo, incorporación de mejoras en funcionalidades y la interfaz.
- **Marzo 2013:** Prototipo final y defensa

Capítulo 3

Estudio de soluciones

Durante el desarrollo de este proyecto, se partió de la idea de incluir la herramienta ideada en el Moodle que se utiliza en la ETSIT. Para ello, se realizó un estudio a fondo sobre Moodle, y las posibilidades que nos permitía para la agregación de nuestra idea.

3.1. Moodle

Moodle es una plataforma de software libre bajo licencia pública GNU¹. Es una aplicación de *e-learning*, o, también conocido como aprendizaje en línea, donde es incluida dentro de los grupos de tecnologías llamadas CMS (*Course Management System*), VLE (*Virtual Learning Environment*) y/o LMS (*Learning Management System*). No hay mucha diferencia entre ellos. VLE puede ser un buen complemento a la formación dentro de un contexto de aprendizaje combinado. CMS tiene un contenido más orientado y permite actualizar fácilmente y cambiar sus recursos y un LMS es mejor en la dirección de la actividad de los alumnos, ya que tiene una ganancia real de capital pedagógico [26].

Martin Dougiamas fue el creador de la plataforma Moodle basándose en sus estudios de Informática y Educación. Dougiamas centró su doctorado en examinar el uso de software abierto para apoyar una epistemología constructorista social de la enseñanza y el aprendizaje dentro de las comunidades basadas en Internet de investigación reflexiva. El constructivismo pedagógico defiende la idea de que el alumno aprende a partir de las construcciones que hace en la mente sobre los temas a tratar y no sobre su transmisión sin cambios a partir del material o recursos proporcionados [25] [27].

Su primera versión apareció en agosto de 2002 con Moodle 1.0. Actualmente está disponible la versión 2.3.3 y hay más de 80 versiones entre la primera y última. Para hacernos una idea del uso a medidas mundiales, en octubre de 2012 contó con un número de sitios registrados y

¹Licencia pública GNU. Moodle tiene los derechos de autor, pero permite que los usuarios puedan copiar, modificar y usar el paquete Moodle. Sin embargo, estos usuarios deben aceptar distribuir el código fuente a otros, no modificar o eliminar la licencia original y los derechos de autor, además de aplicar esta misma licencia a cualquier trabajo que se ha derivado de él (como está reflejado en [24]).

páginas verificadas de 70.793, dando servicio a 63.204.814 de usuarios, más de 6,7 millones de cursos y con más de 1,2 millones de profesores [25].

Filosofía

La filosofía adoptada por Moodle combina las teorías pedagógicas del constructivismo, donde el conocimiento se entiende como una interacción con el entorno, el construccionismo, donde se cree que se obtiene mayor efectividad cuando se construye algo que puede llegar a otros, y el constructivismo social. Trata de juntar los diferentes tipos de comportamientos: separado, conectado y constructivo. Un comportamiento separado es aquel en el que un individuo trata de defender sus ideas por medio de sus razonamientos frente a sus oponentes. El comportamiento conectado se da cuando alguien escucha al que está transmitiendo la información y realiza preguntas para así entender su punto de vista. Y un comportamiento constructivo es cuando alguien acepta ambos puntos de vista y puede elegir el razonamiento adecuado según la situación. Para Moodle, el profesor pierde la característica de fuente de conocimiento, es decir, Moodle ve al profesor más como el encargado de crear un marco donde los estudiantes y él estén conectados de una forma personal y que pueda solventar las necesidades de cada alumno para llegar a los objetivos docentes establecidos en el curso [24].

Especificaciones técnicas

La plataforma Moodle funciona, sin ser necesaria ningún tipo de modificación, sobre la mayor parte de los sistemas operativos como Unix, Linux, FreeBSD, Windows, Mac OS X o NetWare. Los servidores Web más comunes y usados para Moodle son Apache o IIS, pero también está soportado por lighttpd, nginx, cherokee, zeus o LiteSpeed. Para la utilización de Moodle, también se requiere tener instalado PHP, siendo la versión PHP 5.3.2 la mínima soportada. Para guardar toda la información que se expondrá y se utilizará en Moodle se necesita de una base de datos SQL, entre las que se puede elegir: MySQL, PostgreSQL, MSSQL, Oracle o SQLite [25].

Para este proyecto de fin de carrera se ha optado por un servidor Apache, PHP 5.3.2 y MySQL. Todos ellos estudiados en los anteriores cursos de esta Ingeniería. (Para más información ir al apéndice donde se detalla la instalación de Moodle y de todo su software requerido).

3.1.1. Características de Moodle

Moodle es una plataforma que está en constante evolución. Se caracteriza por una serie de servicios: tener un diseño general, permitir una administración del sitio y de los usuarios, así como también de cursos y puede albergar cantidad de módulos como tareas, chat, consulta,

foro, cuestionario, recurso, encuesta o taller [30]. De entre todas ellas sólo vamos a destacar la siguientes:

Diseño general

Como ya hemos comentado, Moodle se creó siguiendo los fundamentos de la pedagogía constructivista social, haciendo de ella una de las herramientas con mayor acogida para los cursos en línea o bien para servir de complemento o apoyo para los cursos presenciales.

Moodle es muy fácil de adquirir e instalar en la mayor parte de máquinas que soporten PHP y tengan una base de datos lista para usar. La interfaz es sencilla, ligera, eficiente, además de tener una seguridad fuerte donde se comprueban los formularios, se codifican las cookies, etc. [30].

Administración del sitio y de usuarios

La administración del sitio corre a cargo de un usuario otorgado con tales privilegios. Se pueden añadir más módulos de los que ya vienen por defecto, pudiendo escoger de la lista que aparece en el apartado de Plugins de la página oficial. En ella, se encuentran diversos tipos de módulos con la seguridad de estar aceptados por la propia organización de Moodle, lo que asegura un correcto funcionamiento y foros en los que plantear las dudas o problemas que surgan en su instalación. Moodle está escrito en código PHP siguiendo un estilo claro y que facilita las modificaciones que cada administrador desee.

La fuerte seguridad de la que se caracteriza Moodle sirve para disminuir al máximo el trabajo del usuario administrador. Permite una gran cantidad de métodos de autenticación por defecto y, además, pueden instalarse otros módulos para permitir el acceso a Moodle desde otras plataformas. El método clásico para la autenticación es por medio del correo electrónico, pero también son soportados otros métodos como LDAP *Lightweight Directory Access Protocol*, IMAP *Internet Message Access Protocol*, POP3 *Post Office Protocol*, NNTP *Network News Transfer Protocol* por medio de los certificados SSL *Secure Sockets Layer* y su descendiente TLS *Transport Layer Security*. También puede ser utilizada como mecanismo de autenticación externo una base de datos que disponga de dos campos como mínimo [30].

Ventajas

Esta plataforma tiene una gran variedad de ventajas [31], entre las que hay que destacar en lo referente a este proyecto:

- Moodle trata de promover la comunicación entre alumnos y profesores o entre alumnos fuera de los límites de las aulas.
- Hace que los usuarios mantengan e impulsen su familiarización con el uso de tecnologías.

- El servicio es soportado por distintos navegadores Web y su uso resulta accesible, confiable, estable, ligero y sencillo.
- Su uso, modificación y gestión son sencillos y con gran variedad de posibilidades para favorecer el desarrollo del estudiante.
- El administrador puede modificarlo y personalizarlo a su gusto para suplir sus necesidades.
- Desarrollado bajo el sello de software libre.
- Otras ventajas serían la alta disponibilidad, escalabilidad, interoperabilidad y seguridad.

Desventajas

La desventaja que más relación guarda con este proyecto es la sensación de aislamiento que puede provocar su uso. Es necesario fomentar e impulsar la comunicación y colaboración entre las distintas partes [31], ya que Moodle, minimiza la interacción con personal docente y entre estudiantes, y ello lleva a una rotura del vínculo que se obtendría de las interrelaciones en los cursos presenciales. Es por ello que este proyecto de fin de carrera se presenta como una solución, puesto que dotando a Moodle de un servicio de red social, donde los alumnos puedan comunicarse entre ellos casi con las mismas posibilidades que en sus redes sociales personales y dedicadas al ocio, en un ambiente universitario y profesional.

3.1.2. Soluciones estudiadas

Modificar el perfil de usuario

Una de las características de Moodle, que nos impulsó a estudiar la plataforma como primera solución, fue que, dentro del Moodle de la ETSIT se tendrían los datos sobre las asignaturas cursadas, superadas y las calificaciones de los usuarios alumnos; puntos bastante interesantes por parte de los profesores que tengan que deliberar qué candidato quieren para su proyecto.

Nuestro primer paso fue editar el perfil de los usuarios de Moodle con más campos. Nuestra idea era incluir nuevos campos opcionales junto con los ya disponibles en Moodle y recopilar toda esa información para el uso de nuestra aplicación. Añadimos campos donde los alumnos pudieran redactar sus intereses, sus conocimientos y experiencia en temas relacionados con las telecomunicaciones y nivel de estudios sobre otros idiomas.

Esta edición del perfil de alumno se realizó por medio del usuario administrador de Moodle [Figura 3.1].

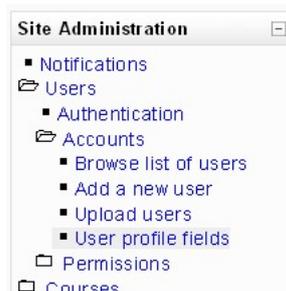


Figura 3.1: Editar/Añadir campos de perfil usuario Moodle

En el apartado de Ajustes → Administración del sitio → Usuarios → Cuentas → Campos de perfil de usuario.

Dentro de este punto, se pueden crear nuevas categorías con diferentes campos relacionados con ellas. Para crear una nueva categoría debemos hacer click en el botón "Crear una nueva categoría de perfil" darle un nombre único y guardala. Los campos de perfil se pueden configurar como diferentes tipos: checkbox, text area, text input, fecha, etc. Pudiendo elegir el campo que más se ajuste a nuestras necesidades. Tras esto, se deberá configurar los siguientes ajustes en función de lo que queremos realizar, pero en general, para cualquier campo nuevo que introduzcamos [49]:

- Un nombre corto para el campo, debe ser único.
- Elegir un nombre para este nuevo campo que sea representativo de la información que debe contener.
- Descripción del campo.
- Elegir si se desea que el campo sea obligatorio, se exigirá al usuario tener escrito este campo de su perfil.
- Decidir si este nuevo campo debe ser bloqueado o no una vez relleno por el alumno.
- Si el valor especificado por el alumno debe ser único o no. Por ejemplo identificación de correo electrónico.
- Podemos elegir si el usuario tiene que crear su propia cuenta en Moodle, podemos hacer que el campo creado conste en la página de registro.
- Quién puede ver este campo. Podemos escoger entre tres opciones de visibilidad: visible para todo el mundo, visible sólo para el usuario (en el caso de información privada) o no visible (si el administrador quiere mantener datos privados de los usuarios).

- Si al haber creado un nueva categoría de campo de perfil, puedes elegir en cuál categoría lo quieres introducir.

Los campos que hemos comentado antes que añadimos como primera prueba, los pusimos opcionales, puesto que, en un principio, deberían estar completados por alumnos de último curso. Además los configuramos para que fueran visibles para todos los usuarios, para que los profesores pudieran llegar a esa información.

En el siguiente punto continuamos explicando nuestra tarea de edición del perfil de usuario alumno para el uso de nuestra herramienta.

Añadir certificados al perfil de usuario

Dentro del sitio web oficial de Moodle podemos encontrar múltiples puglins que pueden resultarnos de mucha utilidad para la realización de nuestro proyecto. De hecho, echando un ojo a la colección de plugins, encontramos uno muy relacionado con el anterior punto. El plugin en cuestión trata de añadir el símbolo de un certificado al perfil de usuario [Figura 3.2].

The screenshot shows the Moodle user profile for Roman Doubush. The page includes a navigation menu on the left with options like 'Home', 'My home', 'Site pages', 'My profile', 'Courses', 'Users', and 'Roman Doubush'. Below the navigation is a 'Settings' section for profile settings. The main profile area displays the user's name 'Roman Doubush', a profile picture, and personal information: Country (Ukraine), City/town (Ivano-Frankivsk), and Email address (roman.doubush@gmail.com). A 'Certificates Wall' section is visible, containing two certificates: one from Adobe titled 'Certified Expert' and another with a shield icon. The page also shows 'Course profiles' and 'First access' information.

Figura 3.2: Ejemplo de perfil usuario con certificados en Moodle [50]

Este plugin se utiliza para mostrar los logros de los alumnos en su perfil [50]. El plugin recoge los datos de este usuario en la página de *CertificateWall.com* y muestra los certificados del alumno en su perfil dentro de Moodle. Si el alumno no tiene ninguno no mostrará nada. El

objetivo a conseguir es permitir que el estudiante o el profesor muestren sus propios certificados, diplomas premios o insignias sobre su educación en su perfil de Moodle. Algo bastante interesante si queremos evaluar de una forma más oficial la formación de un alumno sobre un tema.

Este plugin estaría dentro del tipo de campo de perfil y para instalarlo, el administrador de Moodle debe proceder de la siguiente forma [50]:

1. Descomprimir el archivo del enlace de descarga que se proporciona desde el sitio oficial de Moodle en la sección de plugins.
2. Copiar la carpeta de CertificatesWall al directorio de `/user/profile/field` de la instalación Moodle.
3. Abrir el editor de campos personalizados, en Administración del sitio →Usuarios →Cuentas →Campos de perfil de usuario.
4. Seleccionar CertificateWall en "Crear un nuevo campo de perfil".
5. Configurar y guardar el nuevo campo.

Campo de perfil de inscripción

Este plugin permite unir una estrategia de afiliación a un valor en el perfil de usuario. Se trata de un módulo de matriculación que posibilita al usuario registrarse en un curso presentando los valores adecuados en su perfil de usuario.

El administrador puede utilizar este plugin para determinar el público en algunos cursos añadiendo campos personalizados en el perfil de usuario y ajustando valores ocultos con fines de gestión (estrategia amplia del sitio) [51].

Con este plugin, un alumno se suscribirá a un grupo existente en el momento de matricularse si:

- Hay un grupo definido en el curso.
- Los grupos tienen una contraseña para inscribirse.
- Los grupos sin contraseña no estarán a cargo de esta función adicional.
- Los usuarios pueden optar por aplicar para un grupo al entrar en el curso por primera vez y contesten al formulario de inscripción.

Para su configuración el usuario administración debe seguir los siguientes pasos [51]:

1. Descomprimir el directorio y guardarlo en la carpeta de Moodle *enrol*.
2. Ir a Administración del sitio →Notificaciones para instalarlo en Moodle.
3. Para que funcione correctamente, se necesita que se active este método de matriculación en la administración central de plugins de matriculaciones antes de que se pueda utilizar en los cursos. Por lo tanto se debe ir a Administración del sitio →Complementos →Inscripciones →Administrar complementos.
4. Hacer clic para habilitar el plugin.
5. Si se desea que este método esté por delante de otros, deberemos de modificar su posición en la pila a una superior.

En un principio, se pensó utilizar una combinación de estos tres plugins para la creación de nuestra herramienta. Con un perfil con más campos de interés y permitiendo a los alumnos afiliarse a unos cursos en función de estos campos. Esos cursos serían los diferentes flujos de publicación de ofertas en función de temática o especialización del campo de las telecomunicaciones. En ellos, se podría hacer llegar a todos los alumnos que estén interesados en un área específica la información de las ofertas que dispone la ETSIT.

LinkedIn

Buscando entre los plugins de Moodle se encontró uno relacionado con la red social LinkedIn. Como nuestra primera idea era crear un perfil profesional para el alumno, creímos que este plugin sería muy interesante, puesto que permitía importar datos desde la cuenta del usuario en LinkedIn a Moodle [Figura 3.3].



Figura 3.3: EBloque LinkedIn [48]

Para el correcto funcionamiento de este plugin se necesita de dos bloques: LinkedIn Oath Authentication plugin y el bloque de LinkedIn. Ambos desarrollados por Bas Brands [48], donde para la implementación del bloque de Moodle Oath siguió los ejemplos de autenticación

utilizados por Jerome Mouneyrac. Estos plugins también están publicados y se pueden descargar desde la herramienta GitHub.

La combinación de estos plugins permite que, tras su correcta instalación, el usuario puede iniciar sesión mediante su cuenta vinculada. Para ello, se necesita que el usuario tenga una cuenta en LinkedIn, una clave para la integración entre LinkedIn y Moodle y añadir código personalizado en la herramienta de Moodle.

Para la correcta instalación del cada bloque en Moodle se deben de seguir los siguientes pasos que vamos a señalar.

Bloque LinkedIn auth

El procedimiento que se debe seguir el usuario administrador de Moodle para una adecuada instalación sería [48]:

1. Añadir el código de este plugin dentro del directorio */auth/linkedin*.
2. En la plataforma de Moodle, ir a la página de plugins dentro de ajustes. En la sección sobre autenticación, permitir autenticación por LinkedIn.
3. Visitar la página <https://www.linkedin.com/secure/developer> y añadir una nueva aplicación.
4. Introducir los detalles del sitio web de Moodle, donde el campo más importante es: integración URL. Se deberá utilizar el *wwwroot* de su instalación de Moodle.
5. Una vez se haya recibido desde LinkedIn, introducir la clave de la página de configuración de plugin de autenticación de LinkedIn en Moodle.

Bloque LinkedIn

Para disponer del bloque LinkedIn dentro de Moodle, se deberán realizar los siguientes pasos [48]:

1. Instalar el bloque LinkedIn
2. Ir a la página de ajustes del usuario administrador Siteadmin →Notificaciones e instalar el bloque.
3. Agregar el plugin a la página principal de Moodle.

El procedimiento sería el siguiente [48]

1. El usuario hace un evento de ratón en el nombre de usuario en el botón de LinkedIn.

2. Se abre un *pop-up* que redirige a la página de LinkedIn para pedir su nombre de usuario en la red social de LinkedIn, su contraseña de dicha cuenta y su autorización para dar acceso a Moodle a tu perfil. Si el usuario ya ha iniciado sesión dentro de LinkedIn en otra ventana del navegador, la ventana emergente sólo pide permiso para iniciar la sesión utilizando LinkedIn.
3. Cuando el usuario termina el vínculo de la autenticación en LinkedIn, la página de Moodle se recarga y se redirige al usuario a un formulario pidiendo su correo electrónico.
4. Cuando se hayan completado los pasos anteriores, los usuarios al iniciar sesión de nuevo en Moodle y la información del perfil será mostrada en el bloque de LinkedIn.

La información que se mostrará son los datos siguientes: nombre, apellidos, trabajo actual, país ciudad e imagen de perfil, todos ellos se recuperan de LinkedIn y se presentan en Moodle.

3.2. Mahara

Durante la fase de investigación en las herramientas de Moodle dimos con otra herramienta llamada Mahara que captó nuestra atención por las posibilidades que presentaba.

Mahara es un sistema de redes sociales combinado con un ePortfolio. Un ePortfolio es un sistema en el cual los estudiantes pueden grabar evidencias de su aprendizaje como por ejemplo ensayos. Mahara da forma para que las personas interactúen entre ellas y creen sus propias comunidades en línea. Pero además, es mucho más que sólo un lugar para almacenar archivos, incluye blogs, un constructor de curriculum vitae e integración con Moodle.

Mahara permite un perfil de usuario mucho más completo, no sólo con los campos habituales y el constructor de CV. Los usuarios pueden publicar los blogs que siguen o crear uno suyo dentro de la misma herramienta, pueden compartir vídeos y artículos que crean interesantes con el resto de usuarios. Pueden escribir una breve introducción sobre ellos mismos. Integrando Mahara con Moodle los usuarios podrían disponer de los datos de Moodle en Mahara automáticamente. Por todo esto hizo que consideramos Mahara como una fuerte alternativa para albergar nuestra herramienta, aunque finalmente se descartó por razones que se detallan más adelante.

3.2.1. Integración con Moodle

Dado que las aplicaciones web han crecido en popularidad, un usuario necesita recordar varias cuentas y contraseñas para poder hacer uso de ellas. Para hacer más cómodo el uso entre Mahara y Moodle se utiliza el procedimiento de Single Sign On (SSO). Un usuario inicia sesión en una aplicación y mediante los enlaces diseñados se puede acceder a la cuenta en otra aplicación, sin ser necesaria la creación de otra cuenta, de encontrar un nombre de usuario

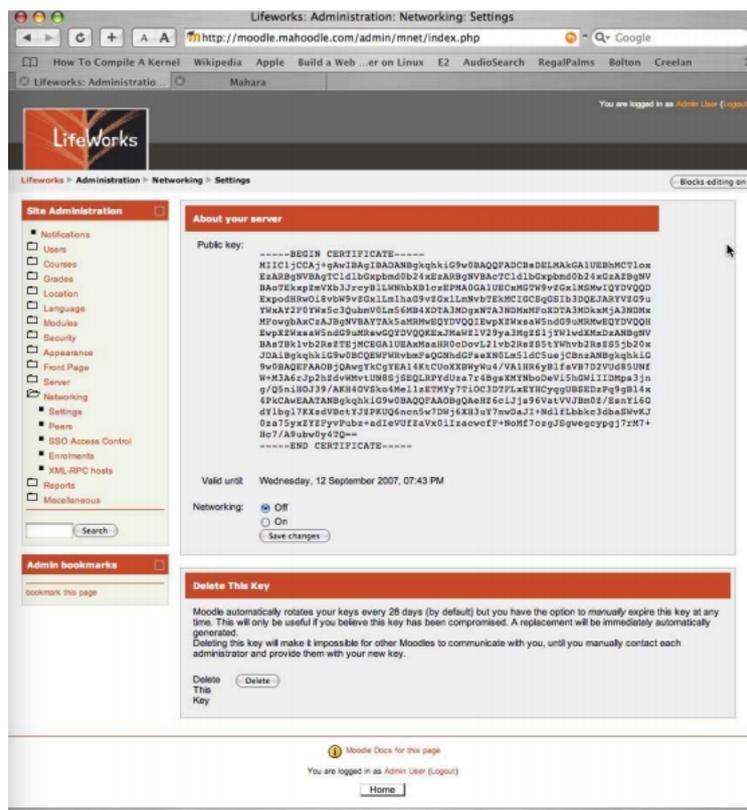


Figura 3.4: Red básica Moodle [52]

único, sin tener que recordar otra contraseña, ni URLs, ni rellenar la misma ficha de perfil en la nueva aplicación [52].

Para llevar a cabo esta integración seguimos una serie de pasos [52]:

1. **Red básica** Teniendo Moodle y Mahara instalados debidamente, iniciar la sesión de administrador en ambas aplicaciones. Desde el usuario administrador en Moodle debemos ir a Administración del sitio → Networking o Red → Ajustes y cambiar el valor de la Red a "On". Guardar los cambios [Figura 3.4] [52].

Antes de continuar configurando Moodle, deberemos habilitar la red en Mahara. Para ello, debemos dirigirnos a Administración del sitio → Networking o red. En esta ventana, la raíz WWW Root debería estar mostrada. Esta es la dirección canónica de su aplicación Mahara puede acceder y que se necesitará introducir en Moodle más adelante. Debajo de WWW, aparece la clave de encriptación pública. Es un archivo de texto aparentemente al azar que comienza con la línea "— BEGIN CERTIFICATE—" y la fecha de caducidad que se observa debajo. Mahara generará automáticamente cuando expire y propagará la nueva clave a sus *peers*. En el desplegable de más abajo para habilitar red lo cambiamos a la opción 'Sí'. El menú desplegable "Auto-registrar all host" lo dejamos con la opción 'No' y guardamos los cambios [Figura 3.5] [52].

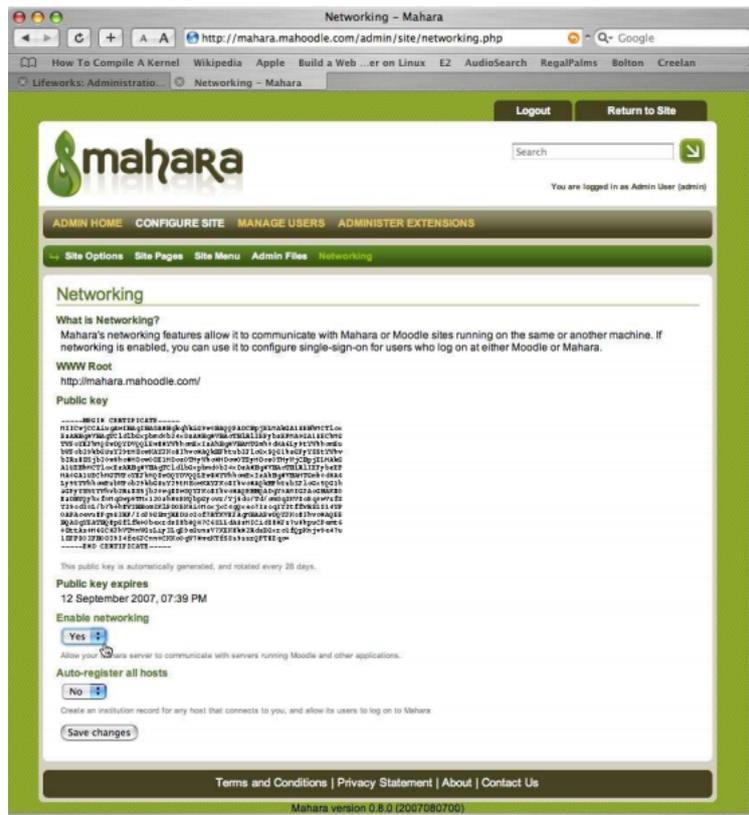


Figura 3.5: Red básica Mahara [52]

2. **Habilitar SSO : Mahara** Seguimos en Mahara. Volvemos a la página de inicio dentro del usuario administrador. Dentro del menú de administración entrar en la sección de Gestión de Usuarios y hacer clic en enlace para Instituciones. Si se está seguro que Mahara sólo servirá a Moodle, podemos hacer clic en Editar para actualizar los detalles para la institución que aparece por defecto. De lo contrario, hacer clic en 'Añadir institución' [Figura 3.6] [52].

El nombre de la institución puede ser cualquier palabra compuesta por letras, sin espacios ni números. El plugin de autenticación no se puede configurar hasta que el registro de la institución se haya realizado, por lo que es ignorado por el momento. Los valores por defecto en los otros campos serán suficientes para empezar. Hacer clic en 'Enviar' la parte inferior de la ventana para guardar esta configuración. Volverá a aparecer la página y el menú del plugin de autenticación se activará [Figura 3.7] [52].

En la ventana que ha reaparecido, se debe elegir *xmlrpc* del plugin de autenticación en el menú desplegable. Hacer clic en el signo más que aparece al lado de este menú para añadir un nuevo método. Se abrirá una nueva ventana que muestra las opciones de configuración para XMLRPC [Figura 3.8] [52].

Dentro de esta nueva ventana el campo de nombre de identidad sirve para identificarla

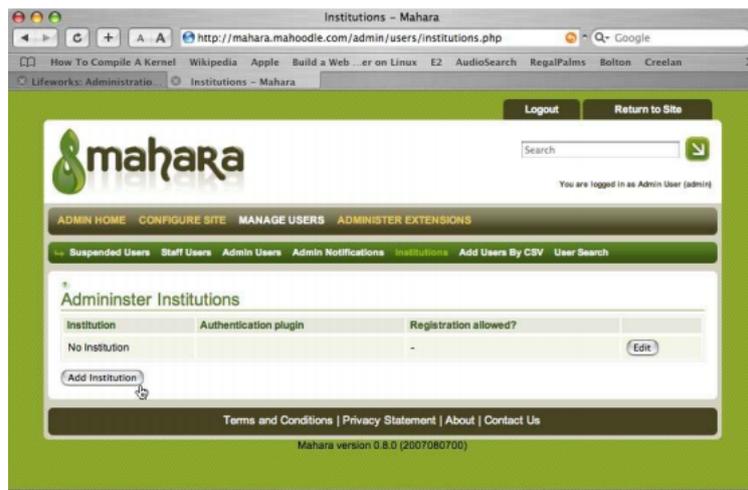


Figura 3.6: Habilitar SSO en Mahara [52]

para el usuario administrador. En el campo de 'Parent authority' debemos mantener la opción de 'Ninguna'. El paso más importante está en el campo WWW Root. En este campo debemos introducir la dirección exacta de Moodle en nuestra carpeta raíz, es decir la misma dirección que aparece en el fichero de config.php de Moodle. A continuación seleccionar en el menú desplegable de más abajo la aplicación que queremos elegir, en nuestro caso Moodle y permitir algunos servicios de red en las casillas de abajo. Debemos seleccionar las opciones 'They SSO in' para permitir a los usuarios entrar en Mahara desde Moodle, con 'Update user info on login' se está pidiendo al servidor de Mahara que compruebe que el usuario no tiene cambiados sus detalles en Moodle cada vez que se conecte y seleccionar también 'We auto-create users' para automatizar el registro de los usuarios dentro de Mahara. Tras estos pasos damos a Enviar [52].

3. **Habilitar SSO : Moodle** Una vez completado el anterior paso, volvemos a Moodle. Nos dirigimos a Administración del sitio → Networking o Red → Peers. Una vez allí, en el apartado 'Añadir un nuevo huésped' se debe introducir el valor de WWW Root del sitio Mahara. Elegir Mahara en el menú desplegable y hacer clic en 'Agregar host'. Moodle obtendrá la clave de encriptación pública del servidor de Mahara y lo mostrará en pantalla [Figura 3.9] [52].

El nombre del sitio se puede editar, sirve para identificar Mahara dentro de Moodle. En esta ventana volvemos a tener un campo parecido al que nos encontramos en Mahara con la clave pública. Guardamos los cambios. Aparecerá la misma ventana pero con otra pestaña en la parte superior. Ir a la pestaña de 'Servicios' para instalar SSI con Mahara. El primer apartado dentro de la lista de servicios es 'Moodle Networked Enrolment' y es relevante sólo para redes con otras plataformas Moodle, por lo que se debe pasar por alto para nuestra finalidad [Figura 3.10] [52].

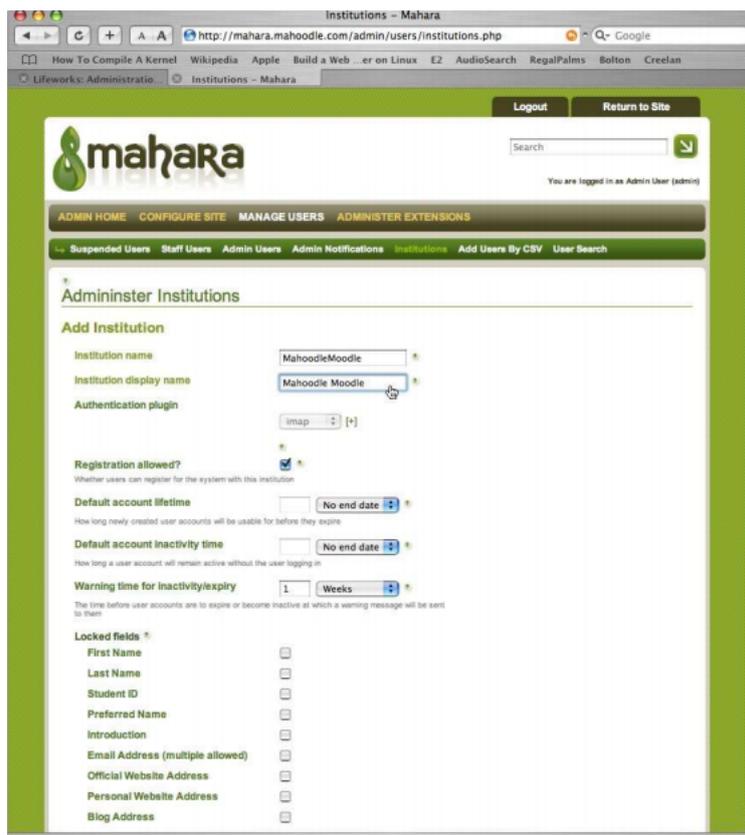


Figura 3.7: Habilitar SSO en Mahara [52]

En la sección titulada como 'SSO (Identity Provider)' marcar la casilla 'Publish'. Moodle va a publicar un servicio a Mahara que identifica a los usuarios de Mahara [52].

En la sección 'SSO Service Provider' hacer clic en 'Subscribe' para que los usuarios de Moodle puedan suscribirse a los servicios que son proporcionados por Mahara. Guardar los cambios [52].

En el panel de Administración del sitio, hacer clic en Usuarios para elegir la autenticación desde el menú. Desde ahí, habilitar el plugin de autenticación de red de Moodle y hacer clic sobre Configuración. En realidad no queremos cambiar los valores de esta ventana, sólo comprobar que nuestro sitio Mahara está aquí [Figura 3.11] [52].

El paso final es ofrecer a los usuarios vagar en el mismo sistema de Clases de Moodle. Elegir Usuarios → Permisos → Definir roles del sitio dentro del menú de administración. Editarlo para el 'Usuario autenticado' [Figura 3.12] [52].

Desplácese hasta la opción 'Roam a un Moodle remoto', haga clic en habilitar y guardar los cambios.

Ya está todo configurado para permitir a nuestros usuarios de Moodle vagar por Mahara. Sólo falta proporcionarles un enlace para llevarlos a Mahara. Volver a la página prin-



Figura 3.8: Habilitar SSO en Mahara [52]

principal de Moodle y hacer clic en 'Activar edición'. Elegir el bloque de 'Servidores de Red' dentro de este menú desplegable de bloques, el servidor de Mahara debería aparecer. Y ya tendríamos el enlace, los usuarios ya pueden disfrutar del uso de Mahara desde Moodle [Figura 3.13] [52].

3.3. Conclusión sobre las diferentes soluciones

En un principio, se pensó utilizar una combinación de los plugins de Moodle para la creación de nuestra herramienta. Con un perfil con más campos de interés, completando la información del usuario en Moodle con la de LinkedIn y permitiendo a los alumnos afiliarse a unos cursos en función de estos campos. Esos cursos serían los diferentes flujos de publicación de ofertas en función de temática o especialización del campo de las telecomunicaciones. En ellos, se podría hacer llegar a todos los alumnos que estén interesados en un área específica la información de las ofertas que dispone la ETSIT. El problema de esta solución es que en la Universidad cuando se quiere agregar un nuevo bloque o plugin al Moodle se necesita un pedido de una forma justificada y no siempre se resuelve favorablemente. Debe existir un gran necesidad real para motivar su modificación.

La solución de Mahara nos motivó para dar a nuestra herramienta un carácter mucho más social. Los usuarios no sólo podrían usar la herramienta para buscar su T.F.G. si no para establecer comunidades online para la resolución de trabajos en grupo y prácticas en las asignaturas de la Escuela. Los usuarios podrían adoptar una actitud mucho más interactiva compartiendo blogs, noticias y videos todo dentro de un marco mucho más informal que en Moodle. Pero, al igual que para las soluciones de Moodle, lo verdaderamente interesante de esta plataforma era

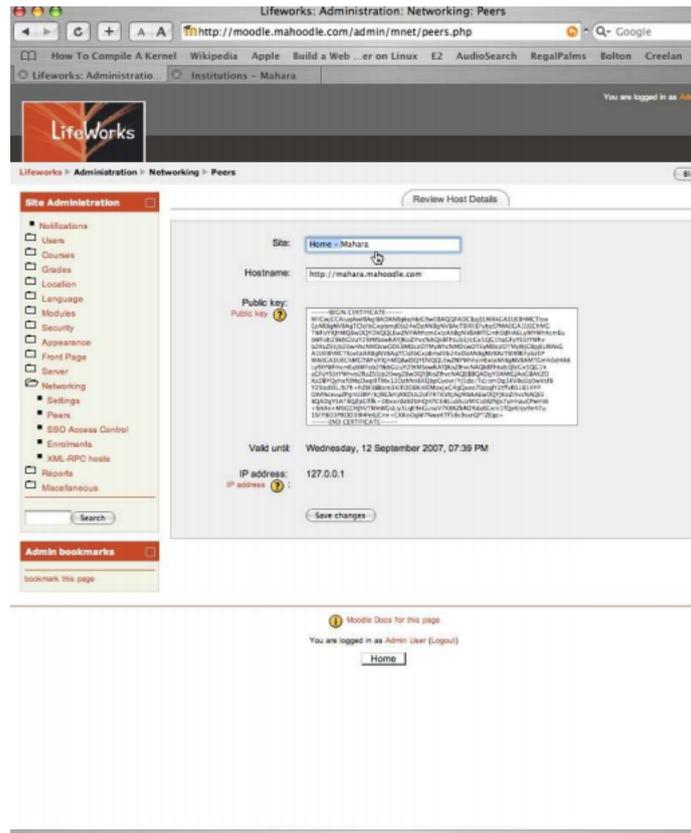


Figura 3.9: Habilitar SSO en Moodle [52]

la cooperación entre Moodle y Mahara, y para ello se debería de pedir a los administradores de Moodle de la Escuela su integración. Por lo que la idea inicial de utilizar Mahara para implantar nuestra herramienta para tenerla de una forma más independiente de Moodle tampoco era válida puesto que de un modo u otro se debía pedir permiso para habilitar esto a la Universidad y quedó descartado. Además, añadir cambios en los códigos de estas dos plataformas no resulta muy fácil. Si es verdad que son modulares, pero cuentan con un código muy extenso y siempre resulta mucho más fácil crear algo nuevo desde tu punto de vista que entender el punto de vista de las distintas personas que se encargaron de la implementación de cada módulo.

Haciendo nuestra aplicación completamente independiente evitamos el proceso de pedir permiso a la Universidad. Si más adelante se prueba su uso y la Universidad lo reconoce, se podría añadir de forma sencilla puesto que decidimos realizarlo en PHP, lenguaje en el que se implementa Moodle.

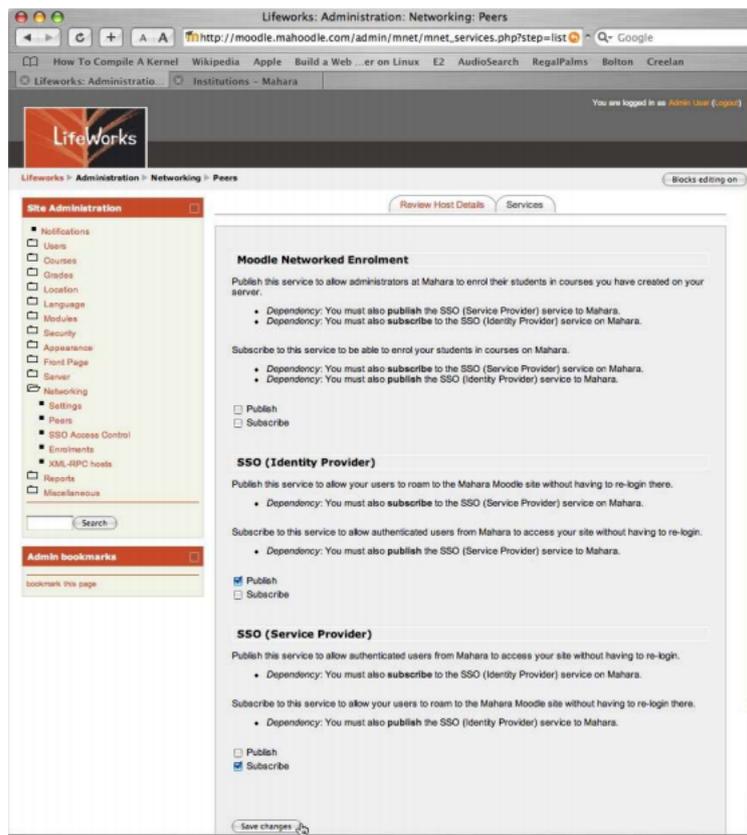


Figura 3.10: Habilitar SSO en Moodle [52]

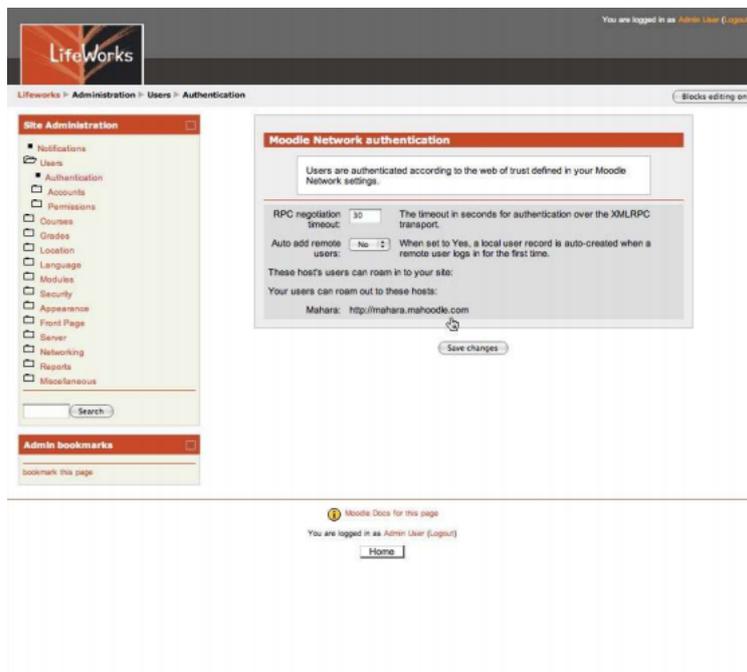


Figura 3.11: Habilitar SSO en Moodle [52]

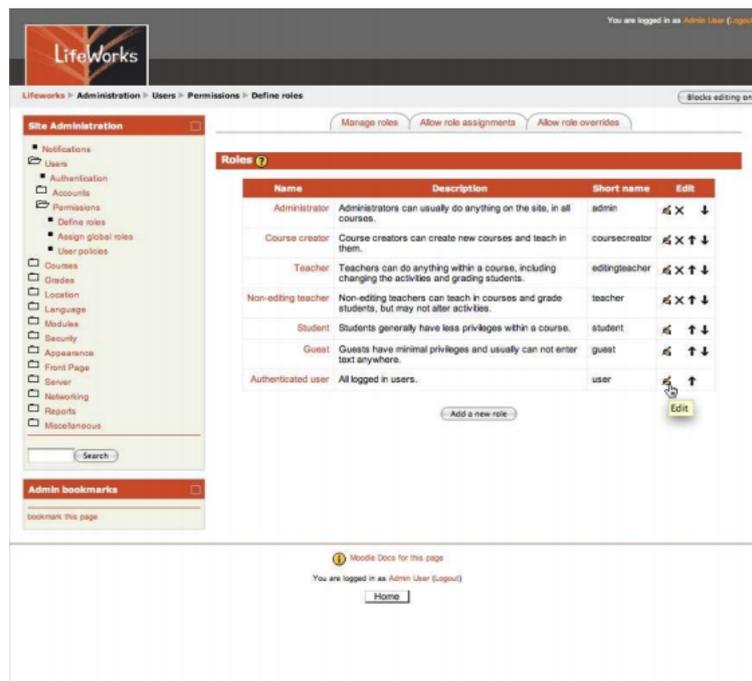


Figura 3.12: Habilitar SSO en Moodle [52]

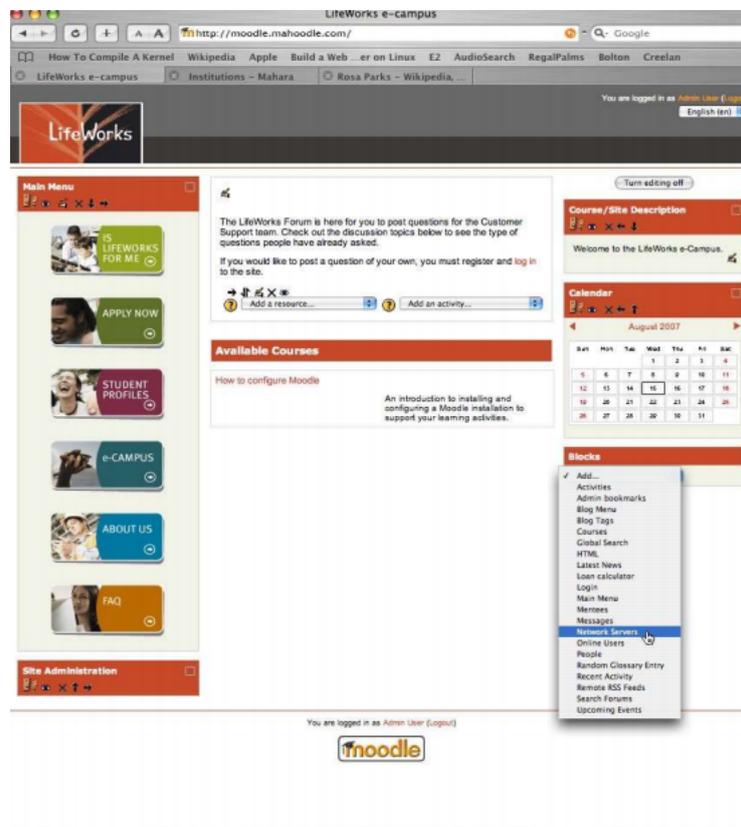


Figura 3.13: Habilitar SSO en Moodle [52]

Capítulo 4

Desarrollo de la herramienta para la concesión de T.F.G.

4.1. Introducción

Esta herramienta se ideó para resolver los problemas que surgen en la ETSIT a la hora de la asignación de los proyectos de fin de carrera a los alumnos de último curso. Durante muchos años el método seguido por profesores y alumnos ha sido de uno a uno, es decir, los alumnos, que tuvieran una idea clara sobre la que querían realizar su proyecto, acudían al profesor que creían más adecuado para ello y se ponían manos a la obra, o, profesores que tenían una idea para un proyecto de fin de carrera hablaban con aquellos alumnos que podían estar interesados, o, simplemente, notificaban vía Moodle esa idea y aquellos alumnos interesados se ponían en contacto con ellos. De esta forma, tanto profesores y alumnos tenían total libertad para elegirse mutuamente como compañeros de trabajo. Este método, digamos extra-oficial, se ha usado durante años y nunca ha provocado problemas. Sin embargo, debido a que el Rectorado de la Universidad ha querido tener más controlado estas decisiones, ya que, muchas veces no tenía constancia de en qué proyectos exactamente estaba involucrado cada profesor y tampoco veía que se le daban las mismas oportunidades a los alumnos, puesto que muchos de ellos no se llegaban a enterar de las todas las propuestas de proyectos, y esta situación nos dió la inspiración suficiente para crear esta herramienta.

Desde el Rectorado de la Universidad decidieron crear un protocolo de asignación de estos trabajos dedicado a todos los alumnos de todas las distintas carreras que se ofertan en esta Universidad, pero en la ETSIT hemos apostado por una herramienta propia de la escuela, dentro de nuestro pequeño círculo en el que más o menos nos conocemos todos. Con el desarrollo de esta herramienta, tanto Rectorado como alumnos verán de forma oficial las ofertas de proyectos de fin de carrera. Los alumnos podrán estar informados todos al mismo tiempo sobre las nuevas ofertas, de qué tratan, las aptitudes requeridas y podrán solicitar aquellas que crean que encajan mejor. Los profesores tendrán más información para escoger al candidato que consideren más

preparado de entre todos los alumnos que aspiren a conseguir una oferta indicada. Desde el protocolo de la Universidad, si un alumno no ha conseguido un acuerdo con un profesor para su trabajo de fin de grado, se le asignará uno de los que queden sin asignar, siendo la condición de asignación la nota media del expediente académico. Como creemos que esto no es una medida del todo justa, puesto que hay alumnos que destacan más en unas materias que en otras, esto puede repercutir negativamente en la elección del trabajo, ya que es muy posible que un alumno acabe realizando un proyecto que no sea de su agrado y eso siempre repercute negativamente en el desarrollo del mismo. Desde nuestra herramienta, a la que hemos llamado *THAIS*, queremos que los candidatos se elijan por sus conocimientos y sean los profesores, quienes conocen a los alumnos de clase, valoren su rendimiento y aptitud en el proyecto a otorgar. Y, por otro lado, el Rectorado estará al tanto de cuáles son los proyectos que lleva cada profesor, además, el proceso queda más transparente y automatizado.

4.2. Especificaciones técnicas

Para la realización de esta herramienta he aprovechado las tecnologías necesarias para soportar Moodle y Mahara: servidor Apache, PHP 5.3.2 y MySQL, por lo que no es necesaria la instalación de ninguna otra tecnología. Esta combinación se conoce también como LAMP (Linux, Apache, MySQL y PHP) y representa casi un estándar en un sistema de infraestructura de internet. Todas estas tecnologías son utilizadas actualmente tanto por la ETSIT como por la Universidad URJC.

4.2.1. ¿Por qué PHP?

La razón principal de usar PHP fue el hecho de que es el lenguaje utilizado en las herramientas Moodle y Mahara. Al realizarlo sobre su mismo lenguaje, sería mucho más fácil añadirlo a éstas si se desease en un futuro. De momento, no vimos oportuno agregar esta herramienta, puesto que es la Universidad (y no la ETSIT), quien regula el Moodle y se muestra bastante reticente a la hora de añadir nuevos módulos. Y, puesto que esto es la realización de un proyecto de fin de carrera, y existe la posibilidad de que, después de hacer pública nuestra herramienta, no hubiera recibido la aceptación esperada por parte de alumnos y profesores, creímos que era mejor mantenerla como una herramienta independiente, aunque lo desarrollamos pensando en una futura incorporación.

A pesar de que PHP no es un lenguaje que haya estudiado durante la carrera, y lo más normal hubiera sido que la desarrollara en Python, por ejemplo, no he lamentado esta decisión, ya que me ha permitido aprender este lenguaje a fondo, y sumado a los conocimientos sobre otros lenguajes que he recibido durante estos años, puedo decir que tengo una amplia preparación en los lenguajes de programación más usados en Internet: PHP, Python y JAVA (Figuras 4.1, 4.2).

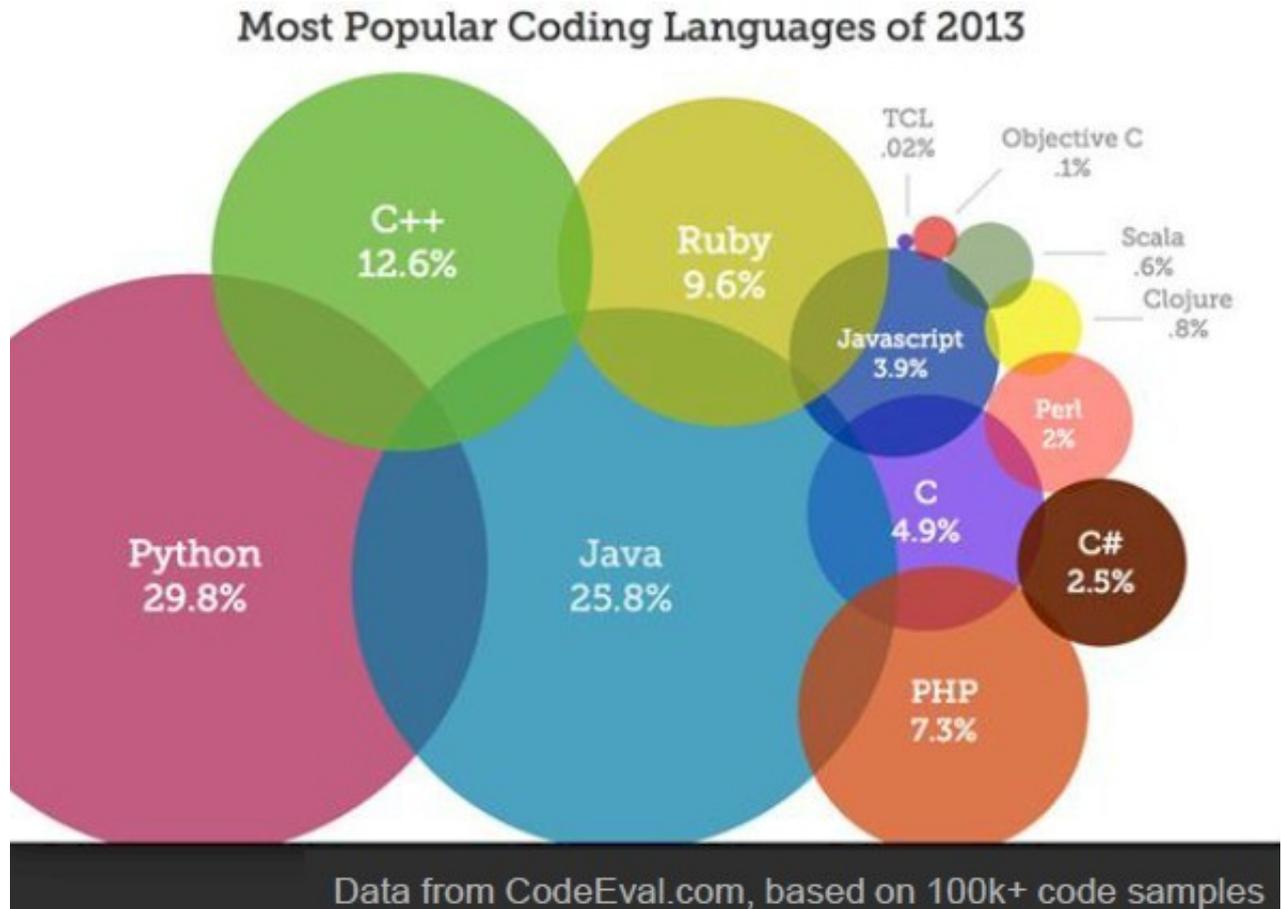


Figura 4.1: Lenguajes de programación más populares en 2013 (Datos sacados de *CodeEval.com* basados en más de cien mil muestras de códigos). Fuente [32]

Entre las características que se pueden resaltar de PHP, señalamos [34]:

- Es un lenguaje de propósito general ampliamente usado, siendo especialmente adecuado para desarrollar webs.
- Puede ser fácilmente embedido en páginas HTML, lo que hace que sea sencilla la tarea de convertir un sitio web estático en uno dinámico.
- Las webs basadas en PHP son fáciles de mantener, mejorar y actualizar.
- PHP es multiplataforma, permite el funcionamiento de diversos sistemas operativos.
- Existe una enorme comunidad de PHP, debido a su gran uso, hay muchos desarrolladores, usuarios y compañías que buscan programadores.
- Sus cualidades más señaladas son la estabilidad, la flexibilidad y la velocidad.
- PHP tiene varias extensiones y es extremadamente escalable.

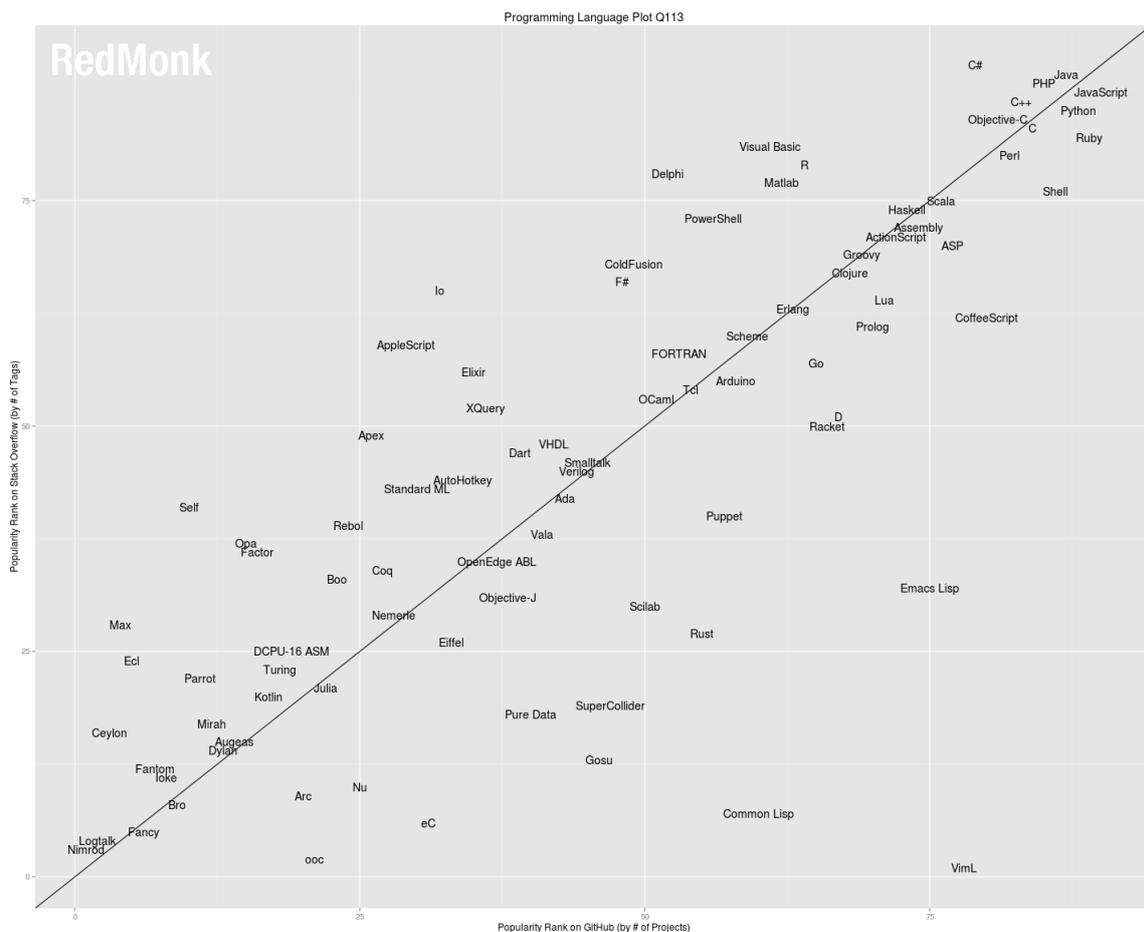


Figura 4.2: Popularidad de los lenguajes de programación. Fuente [33]

Entre los ejemplos más destacados que podemos encontrar de sitios web creados en PHP, los más importantes serían *Wikipedia.org*, *Facebook.com* y *Wordpress.com*.

Pero PHP también tiene algunas desventajas como las siguientes [35]:

- A pesar de su flexibilidad y su naturaleza abierta, para un proyecto complejo exige un cierto grado de experiencia. Aunque es fácil de aprender se requiere un tiempo suficiente para estudiar las ventajas que nos presenta y elegir las herramientas más adecuadas para nuestro proyecto.
- Los scripts para la mayoría de las tareas suelen ser fáciles de implementar, pero cuanto más complejos los hagas, más lento funcionarán.
- Como es un lenguaje de *scripting*, los códigos no se compilan.

4.2.2. HTML5 Y CSS3

Aparte de las tecnologías citadas anteriormente, en este proyecto también se puede encontrar las tecnologías recientes: HTML5 y CSS3 (ver figura 4.3). Estas técnicas eran relativamente nuevas para mí, puesto que durante estos años de estudio, no había trabajado con ellas. Las únicas técnicas que había usado con anterioridad fueron sus predecesoras: HTML y CSS. Obviamente, ya que conocía y había usado en prácticas dichas versiones anteriores, no me ha resultado muy difícil habituarme a ellas, aunque sí que han incorporado importantes cambios respecto de una versión a otra. HTML5 y CSS3 han integrado multitud de nuevas funciones que dotan de un gran dinamismo a las páginas web haciéndolas más interactivas con el usuario si cabe, haciéndolas parecer más una aplicación que un documento.



Figura 4.3: HTML5 y CSS3. [53]

Nosotros nos hemos centrado más en sus avances en la parte interactiva que en sus novedades respecto a la estructura de las páginas. Explicaremos más adelante, en detalle, cómo hemos trabajado con estas tecnologías.

4.3. Desarrollo de la herramienta web

4.3.1. Descripción de la herramienta

La aplicación implementada para la concesión de Trabajos de Fin de Grado (T.F.G.) consta de tres perfiles de usuarios distintos y unas tablas de datos para su correcto funcionamiento.

■ Base de datos

Las tablas que forman parte de la base de datos que hemos creado para esta herramienta son: `pre_registro.sql`, `usuarios.sql`, `ofertas.sql`, `candidatos.sql`, `asignaciones.sql`, `aceptaciones.sql`, `visitas_ofertas.sql`, `cvs.sql` y `fotos.sql`. En la figura 4.4 se puede ver un diagrama de las tablas de la herramienta THAIS. Se trata de un *Entity Relationship Model* de la base de datos de THAIS, conseguido con el programa MySQLWorkbench.

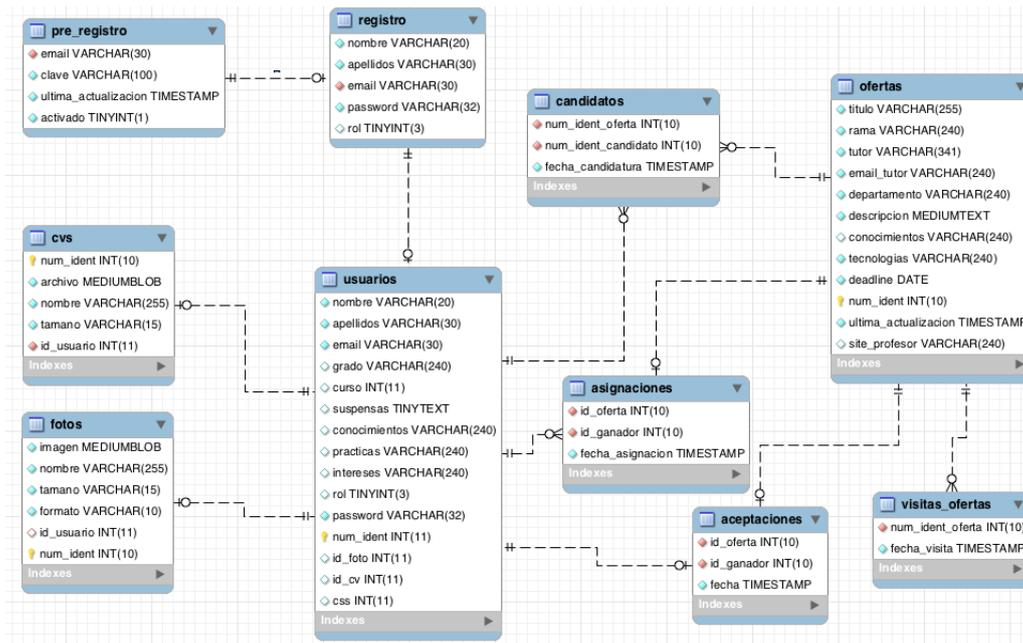


Figura 4.4: Entity Relationship Model de la base de datos de THAIS

- **Pre-registro:** La primera tabla en MySQL, `pre_registro.sql`, se utiliza para guardar las peticiones de creación de cuenta en nuestra herramienta al hacer el pre-registro. En ella almacenamos el email, la clave de activación, fecha de actualización y un valor *booleano* para comprobar si está activado o no.
- **Registro:** En el formulario de la dirección que se envía por correo electrónico tras el pre-registro, se piden unos datos mínimos para la creación de un usuario en la herramienta. Estos datos son el nombre, los apellidos, el email (que ya se mostrará introducido) y la contraseña para el acceso a la cuenta en THAIS. Estos datos se pasan a la tabla `usuarios.sql`.
- **Usuarios:** La tabla `usuarios.sql` sirve para guardar los datos de los usuarios que pedimos en la herramienta. Como puede verse en la figura 4.4, contiene más campos que en el registro, como: grado, curso, conocimientos, asignaturas pendientes, etc.
- **Ofertas:** En `oferta.sql` guardamos toda la información de los trabajos y proyectos. Se piden los campos se observan en la figura 4.4.
- **Candidatos:** En `candidatos.sql` guardamos la identidad de las ofertas y la ID del usuario que ha enviado la solicitud y la fecha en la que envió dicha solicitud.
- **Fotos de perfil y CV:** En las tablas `cvs.sql` y `fotos.sql` incluimos los datos necesarios cuando un usuario sube el archivo de su foto de perfil y su currículum vitae. Guardamos tanto la información binaria como ID de usuario, formato del archivo, tamaño y nombre.

- **Asignaciones:** En la tabla de asignaciones.sql almacenamos el número de identidad de la oferta y del candidato, además de la fecha cuando un profesor selecciona a un alumno para su realización.
- **Aceptaciones:** La tabla aceptaciones.sql, con los mismos campos que la anterior a excepción de la fecha, que en este caso, esa fecha que guardamos es cuando el alumno acepta la oferta que en al que ha sido elegido.

Cada una de estas tablas se necesita en la ejecución de las diferentes páginas.

■ Roles de usuario

Los distintos perfiles existentes son: perfil alumno, perfil profesor y perfil administrador. Cada uno de estos perfiles cuenta con una página de inicio en la que se muestran todas las ofertas existentes y vigentes ordenadas por defecto por fecha de creación, es decir, las ofertas más recientes aparecerán en las primeras posiciones de la tabla. En ella se puede seleccionar sólo las ofertas por una determinada rama /especificación o buscar aquella oferta que se desee. Además de éstas, estos tres perfiles tienen en común la página donde se ve el historial de ofertas de TFG que ya se han concedido y la página para cambiar la contraseña de la cuenta. En la página que contiene el historial de TFG, todos los usuarios pueden ver la información de la oferta en detalle y el nombre de la persona que la realizó. Los perfiles de alumno y profesor también cuentan con una página de notificaciones sobre las distintas noticias dentro de la herramienta relacionadas con su perfil. Aparte de éstas, cada perfil tiene distintas páginas según su rol:

- **Alumno:** En el perfil de usuario con rol alumno, el estudiante cuenta con un perfil donde podrá rellenar con su información personal y podrá editar dicho perfil. La información que el usuario publique será vista por el profesor de la oferta que haya solicitado y utilizada para que deliberar en su decisión de elegir el candidato idóneo. En este proyecto se ha querido tener mucho cuidado con los problemas de privacidad, por lo que los alumnos sólo pueden ver su perfil y no los perfiles de los demás alumnos. Además, tampoco pueden ver qué alumnos han solicitado las distintas ofertas. Se diseñó de esta manera, puesto que el perfil que hemos creado es un perfil que contiene información personal del alumno en lo referente a su progreso académico y muchos alumnos podían mostrarse reacios a enseñar públicamente esos datos. Por otro lado, tampoco creímos que fuera oportuno mostrar a los alumnos qué estudiantes estaban postulando a los trabajos para que no hubiera ningún tipo de rivalidad. De esta forma, el alumno sólo optaría por aquellos trabajos en los que se vea libremente interesado y capacitado; no influido por otros factores secundarios.

- **Profesor:** El perfil de profesor dispone de una página para crear nuevas ofertas. Esta página es un formulario en la cual se pide la información necesaria para describir lo más concretamente la oferta de trabajo de fin de grado, para que los estudiantes puedan ver de forma clara y directa de qué trata el trabajo. Para dichos campos, tomamos como ejemplo el formato actual de los anuncios de proyectos de fin de carrera, pero añadimos algunos campos más que creímos que pueden resultar de interés a los alumnos y definir, aún más si cabe, la temática del trabajo. Los profesores pueden ver en cada oferta los alumnos candidatos y su perfil. Planteamos estas acciones del rol de profesor de esta manera, porque consideramos que podía resultar de gran utilidad a la hora de crear futuras ofertas que los profesores pudieran ver el tipo de perfil de alumno que aspiraba a un tipo de ofertas. Así conseguir llegar a aquellos perfiles de alumnos que más se deseen para un trabajo u otro. Sólo el profesor que sea el tutor de la oferta podrá otorgar, o no, la oferta a un candidato, como es obvio. Asimismo, el tutor de una oferta es el único que tiene permiso para editar o borrar la oferta, ningún otro profesor puede modificar ofertas que no sean suyas. El usuario profesor no tiene una página de perfil dentro de nuestra herramienta, puesto que consideramos que no tenía mucho sentido, porque ya disponen de un perfil y una página personal dentro del Moodle de la escuela y en la página de la Universidad, y resultaría bastante incómodo para ellos tener que rellenar otro perfil más con la misma información. Igualmente, como la herramienta está enfocada a la ETSIT, los alumnos conocen a sus profesores de asistir a clase y saben a qué ámbito de las telecomunicaciones se dedica cada uno y la filosofía que seguirán los proyectos que escojan.
- **Administrador:** El usuario administrador es, sin lugar a dudas, el usuario que tiene más acciones dentro de la herramienta que presentamos. Puede crear nuevas ofertas, editar y borrar cualquier propuesta de proyecto de fin de carrera, seleccionar a un candidato, desotorgar a un candidato ya seleccionado de una oferta o borrar la candidatura del alumno en esa oferta. Puede resetear las contraseñas de cualquier usuario (incluida la suya), puede determinar el rol de cada usuario y eliminar usuarios. Los usuarios que se crean, por defecto se establecen por defecto con el rol de alumno. Cuando un profesor se registre, deberá informar al administrador de la herramienta para que le dé permisos de profesor. Para ello, tendrá que indicar al administrador del sitio la dirección de correo electrónico que utilizó para crear la cuenta en nuestra aplicación. Este procedimiento, para decretar el rol de los usuarios, supone aumentar la carga de trabajo del usuario administrador, pero lo pensamos de esta manera, porque es muy difícil saber con antelación quién está detrás de un ordenador cuando se está realizando el registro y nos pareció la mejor solución ante ese problema de se-

guridad. Este usuario puede designar el rol de otro usuario a los valores de: alumno, profesor o administrador. La única acción que no puede hacer el administrador es editar el perfil de un alumno, ya que pensamos es esa acción vulneraría la privacidad del usuario, puesto que estamos hablando de datos personales. En cambio, sí que dimos permisos a este usuario para seleccionar el mejor candidato, editar, borrar o crear nuevas ofertas, pero como, obviamente, este usuario estará representado por alguien de confianza dentro de la ETSIT, como por ejemplo, alguien que forme parte del profesorado, suponemos que sólo utilizará ese poder para casos puntuales debidos a un error en el sistema, red, software, etc. En resumen, el trabajo de este usuario se basa en asignar los roles de los profesores adecuadamente, depurar el sitio de ofertas muy antiguas (si se desea) y, si se da el caso de algún error, modificar la información de algún trabajo, sumando todo ello, creemos que estas responsabilidades forman un volumen de trabajo adecuado, ya que designar el rol a los profesores, es una acción puntual, eliminar antiguas ofertas, en caso de que se desee liberar la base de datos de las ofertas anteriormente publicadas, o, editarlas, será requerido sólo por motivos de posibles errores ocasionales.

En el apéndice, el lector encontrará mayor información sobre el funcionamiento de la herramienta ha nivel usuario. Hemos creado un manual de usuario para los distintos roles.

4.3.2. Modelo Vista Controlador (M.V.C.)

Para el diseño de la herramienta me he basado en el patrón Modelo Vista Controlador (M.V.C.). El patrón M.V.C es un patrón de diseño de arquitectura de software construido con el fin de separar la capa de presentación de los datos de los métodos que interactúan y transforman con ellos. En teoría, un sistema que siga el modelo M.V.C. completamente debería permitir a un desarrollador de front-end y a un desarrollador de back-end trabajar en el mismo sistema sin interferirse o compartir ficheros. Se considera por este motivo una buena práctica el seguir en cierta medida este patrón de diseño al desarrollar aplicaciones web.

El modelo es el nombre que se le da al almacenamiento permanente de los datos de los objetos usados en el sistema. Debe además permitir un acceso de lectura y escritura sencillo a los datos. En nuestra herramienta, el modelo vendría representado por las tablas de la base de datos que representan los diferentes objetos del sistema (usuario, oferta, foto, CV etc.) y por las clases PHP que proporcionan métodos accesorios a ellas.

El controlador es el encargado de gestionar las interacciones del usuario y de realizar las tareas de validación de formularios. Toma los inputs de los usuarios y ejecuta las correspondientes acciones sobre el modelo en función a éstas. En nuestra aplicación, en términos de ficheros, estaría compuesto por los distintos gestores (GestorOfertas.php, GestorFotos.php etc.),

los ficheros de validación (`validadorFormOferta.php`, `validadorFormularios.php`), los ficheros de transformación y control de flujo (`EnviadorEmails.php`) y la parte de código PHP de las páginas del sistema. Entre las operaciones que puede enviar el usuario están: editar los datos del perfil (en caso de alumno), editar la información de una oferta (en caso de profesor o administrador), cambiar la contraseña de la cuenta etc.

La vista está compuesta por los ficheros HTML y sus correspondientes ficheros CSS (en nuestro caso, páginas como `inicio.php` y sus tres posibles estilos CSS).

Pongamos un ejemplo para que quede claro el funcionamiento del modelo M.V.C. a través del caso de un usuario que desee editar su información de perfil. El usuario rellenará un formulario HTML, dado por la vista. Todos los nuevos datos que el usuario introduzca son captados por el controlador o gestor, el cual realiza la función de validación comprobando que son apropiados. Una vez el gestor haya comprobado la validez de los datos, éste realizará toda la lógica para añadir o modificar los datos en el objeto usuario de la base de datos de la herramienta (modelo). Por último, la vista muestra los datos que están finalmente guardados en el modelo al reflejar en pantalla los datos del perfil del usuario.

Por último, he de hacer notar que en un sistema que siga al pie de la letra el modelo M.V.C. los ficheros de presentación estarían completamente separados de los ficheros de gestión o control. No obstante, esto ocurre raras veces en la práctica y así, en nuestro caso, existen ficheros que comparten ciertas funciones de controlador y asimismo contienen código HTML. No obstante, cuando esto ocurre las partes de vista y controlador están bien delimitadas dentro del fichero.

4.3.3. Explicación de la herramienta

En esta sección explicaremos en detalle la herramienta THAIS. Para ello, describiremos paso a paso el funcionamiento de cada una de las páginas. Empezaremos por orden, entre el registro de un nuevo usuario y una vez dentro de nuestra aplicación. Explicaremos primero los partes comunes de cada página y después entraremos en detalle en las diferencias que tiene cada rol. Por último, comentaremos las partes del CSS y Offline.

Pre-registro, registro e inicio de sesión

La primera página que nos encontramos en nuestra herramienta es, lógicamente, un formulario para iniciar la sesión de usuario. En esta página vemos el formulario para iniciar sesión junto con el torrente de *tweets* de la cuenta en *Twitter* de nuestra aplicación.

Si el usuario no está registrado puede crear una cuenta haciendo clic en el enlace *Registrarse* que aparece justo debajo de este formulario de inicio de sesión [Figura 4.5].

Figura 4.5: Formulario inicio sesión en THAIS

- **Pre-registro:** Este enlace lleva a una página con otro formulario de pre-registro, parecido al de Iniciar sesión, pero en este sólo piden una cuenta de correo electrónico [Figura 4.6].

Figura 4.6: Formulario pre-registro en THAIS

La cuenta de correo introducida se envía al Gestor de Usuarios, que este a su vez reenvía al validador de formularios, para saber si se trata de una cuenta de correo legítima. Una vez hecha esta comprobación, el Gestor de Usuarios es el encargado de comprobar que no existe un usuario dentro de THAIS URJC utilizando ese email. Si todo ha ido bien, después de que el usuario haya enviado su dirección de correo, le llegará un mail, con un enlace a otro formulario donde podrá finalizar el registro. Este enlace tiene como parámetro una clave de activación, que se crea cuando ha pasado todas las pruebas del validador y gestor. Esta clave junto con el email y la fecha de pre-registro se guarda en la base de datos `pre_registro.sql`. La razón por la que creamos esta clave es por seguridad, para que ninguna otra persona pueda suplantar a este usuario en el registro, por eso la incluimos como parámetro en la URL que enviamos en el correo tras el pre-registro junto con la dirección de correo enviada. La forma en la que obtenemos esta clave es así:

```
$key = sha1(mt_rand(100000,999999).$email_new_user);
```

Como un número aleatorio concatenado con la dirección de correo y todo ello pasado por el mecanismo de seguridad SHA-1. La dirección a la página de registro que enviaremos por correo electrónico al usuario se crea de la siguiente forma:

```
<a href="' . $url_publica . ' /thais/formulario_registro.php
?email=' . $email_new_user . ' &key=' . $key . ' ">
```

- Registro:** Esto lleva a otra página con un formulario para el registro. Lo primero que se comprueba es que la clave y el correo pasados por parámetros en la URL son los mismos que los que hay almacenados en la tabla `pre_registro.sql`. En caso de no coincidir se redireccionará al formulario de pre-registro. Si un usuario no termina el procedimiento de registro, es decir, si tras completar el formulario de pre-registro, no realiza el registro, no recibe el correo con la clave de activación o la clave de activación da error, el usuario puede volver a realizar el pre-registro, donde se borrarán sus datos previos y se crearán unos nuevos, con el mismo procedimiento. Se reseteará su pre-registro. Ya en la página de registro se pide al usuario que facilite datos como su nombre, apellidos y una contraseña que deberá introducir dos veces [Figura 4.7].

Figura 4.7: Formulario registro en THAIS

Todos estos datos se envían a un Gestor de Usuarios, que divide la información en bloques y realiza consultas a un validador de formularios para comprobar si los datos introducidos son adecuados. Cada uno de estos campos son comprobados por `ValidadorFormularios.php`. En nombre y apellidos hacemos una comprobación tal que sólo contenga letras y espacios, sin números, símbolos o caracteres de puntuación. Para el email, como hemos dicho antes, comprobamos que tiene la estructura de una dirección de correo electrónico, (`xxxxx@yyyy.zzz`), y puede contener números y caracteres especiales, pero no espacios. Esto lo conseguimos mediante la siguiente instrucción:

```
return (preg_match('/^[a-zA-Z]+([\.\.]?[a-zA-Z0-9_-]+) *
```

```
@ [a-z0-9]+ ([\.-]+ [a-z0-9]+) *
\.[a-z]{2,240}$/' , $email));
```

En el campo de la contraseña validamos que su longitud es mayor que cuatro y que ambas contraseñas pedidas (original y repetición) coinciden. Una vez validadas estas entradas, el validador devuelve una variable booleana con valor *true*, que sirve de orden para que el Gestor cree el nuevo usuario en la base de datos. Destacamos la manera en que guardamos la contraseña es mediante los mecanismos de seguridad: *MD5* y *SHA-1*. El algoritmo *SHA-1* (Secure Hash Algorithm, algoritmo de Hash Seguro) es un sistema que realiza funciones hash criptográficas. *MD5* (Message-Digest Algorithm 5, Algoritmo de Resumen del Mensaje 5) es un mecanismo de reducción criptográfico. Ambos algoritmos son ampliamente usados en aplicaciones web. Para almacenar las contraseñas de los usuarios de nuestra plataforma primero ejecutamos el algoritmo *SHA-1* de la clave introducida y a posteriori, añadimos a su salida el email del usuario y ejecutamos sobre este mensaje *MD5*. La salida tras *MD5* la almacenamos en nuestra base de datos. Cuando un usuario introduce su contraseña en el formulario de inicio de sesión, rehacemos este procedimiento y el resultado obtenido lo comparamos con el que tenemos almacenado en su registro para comprobar que el usuario que intenta entrar es quién dice ser. Todos los cambios de contraseña que dispone la herramienta se hacen siguiendo este mismo esquema.

- **Inicio de sesión:** Tras insertar todos los campos introducidos en el registro, este nuevo usuario se crea con el nombre, apellidos, contraseña, correo electrónico (que además es nombre de usuario en nuestra herramienta) y un número de identidad. Este número de identidad resulta de gran utilidad para tener una cuenta del número de usuarios que dispone nuestra herramienta y también para identificar al usuario cuando realice una modificación de sus datos. Si el registro ha ido según lo planeado, el usuario directamente puede introducir la dirección de la cuenta de correo utilizada durante el procedimiento de registro y su contraseña. Si durante el registro ocurre algún problema, se informará al usuario por medio de avisos impresos en la página. Si el usuario se olvida de finalizar el registro, simplemente acudiendo a la página del formulario de pre-registro y volviendo a introducir su correo, se le enviará de nuevo el correo electrónico con el enlace de activación. Si, desde un principio, estamos ya registrados, simplemente introduciríamos la cuenta de correo electrónico que utilizamos para crear nuestra cuenta y la contraseña para iniciar la sesión [Figura 4.5]. Después de la página con el formulario de inicio de la sesión, antes de enviar al usuario a la página de inicio, pasamos por una página antes, en la cual, creamos la sesión y comprobamos el rol de usuario para redirigirnos a la página de inicio de alumno, profesor o administrador. Como dijimos anteriormente, todos los usuarios se crean por defecto como alumnos. Si alguien quiere cambiar su rol, deberá ponerse

en contacto con el administrador del sitio, para que éste cambie sus permisos.

En estas páginas con los formularios de inicio de sesión, pre-registro y registro, el CSS que se ejecuta es el CSS por defecto, también llamado clásico en la aplicación. Dentro de la herramienta, el usuario podrá elegir el CSS que más le guste, y disfrutará de él en todas las páginas dentro de ella. Pero para el pre-registro, registro o inicio de sesión siempre aparecerá el CSS clásico. Esto es así, ya que no hay mecanismo que nos diga a priori qué usuario va a abrir su sesión para cargar el CSS que tiene elegido. Si el usuario empezara su sesión siempre desde el mismo ordenador con la misma dirección IP, se podría manejar mediante las cookies, pero como eso es casi del todo improbable hemos descartado esa solución. Un usuario puede iniciar sesión desde el ordenador fijo de su casa, desde su portátil en cualquier lugar con distintas redes, desde un ordenador de la Escuela, de la biblioteca, de los laboratorios, etc. Hoy en día el usuario es un usuario que se mueve. Esa solución podría haber funcionado años atrás, pero tampoco habría sido muy efectiva, puesto que al cerrar sesión provocamos la expiración de la cookie y la eliminamos.

Tras estos pasos, llegamos a la página de inicio. El contenido de ella varía un poco dependiendo del rol del usuario.

Página de inicio

Lo primero que el usuario ve son los respectivos botones para seguir a la herramienta en las diferentes redes sociales que dispone. Éstas son Twitter y Google+, donde al usuario le llegarían notificaciones de nuevas ofertas. Además, disponemos de un archivo RSS, que explicaremos más adelante. También, nos encontramos con un menú desplegable de selección, donde el usuario podrá filtrar las ofertas por nombre de especialización, una barra para buscar una oferta, la tabla de las ofertas vigentes y por último, un menú desplegable donde elegir el tema de estilo que más guste [Figura 4.8].

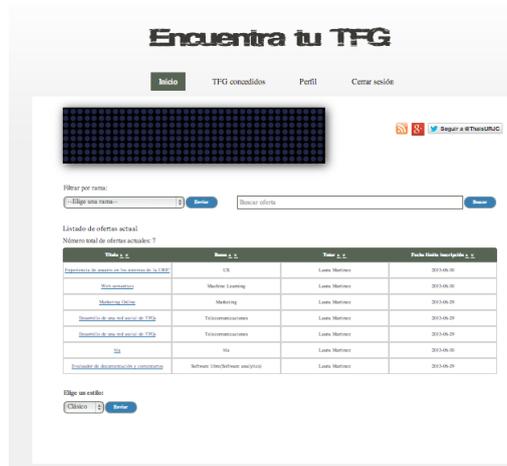


Figura 4.8: Página de inicio en THAIS (Alumno)

La primera vez que carguemos esta página aparecerá en la parte de arriba a la izquierda una imagen de bienvenida que se irá desapareciendo [Figura 4.9]. Lo referente a esta imagen se explicará en el apartado de estilo.



Figura 4.9: Imagen de bienvenida en THAIS

El archivo RSS es un archivo PHP con XML empotrado. En él mostramos una breve descripción sobre las ofertas existentes en toda la base de datos de la herramienta, ordenadas por fecha límite de inscripción. Para ello, creamos un script en PHP para consultar a la base de datos y crear un bucle que generara la salida XML. En esta salida mostramos el título de la oferta, su descripción y su fecha máxima de solicitud. Además añadimos la fecha de la última construcción del fichero XML. Este formato de XML permite compartir contenidos sin que sea necesario el uso de un navegador, con un software concebido para leer la información que contiene, aunque también se puede acceder a sus contenidos a través de un navegador. Las últimas actualizaciones de los navegadores más utilizados hacen posible su lectura sin necesidad de un software adicional.

En la barra de menú desplegable para elegir rama, el usuario escoge una especialización por la cual filtrar la tabla de ofertas a mostrar [Figura 4.10].



Figura 4.10: Selección de rama en inicio

Las especializaciones que aparecen en este desplegable son el resultado de hacer una consulta a la base de datos sobre las distintas ramas de las ofertas que se encuentran almacenadas. Esta consulta se hace directamente sobre MySQL, por medio de la orden:

```
$consulta_rama= "select distinct rama from ofertas where
                 estado='No decidido' ";
```

y genera como salida una tabla con todas las ofertas vigentes y ordenadas de más reciente a menos, en fecha de creación. Esta tabla puede ordenarse por título, tutor o fecha límite de inscripción por medio de las flechas que aparacen al lado de estos campos en la primera fila de la tabla que define cada columna. Cuando decimos que se ordene una tabla en función del

título, tutor o rama, estamos organizando los registros por orden alfabético derecho o inverso. En cambio, cuando organizamos los registros por fecha límite de solicitud el orden se establece por mayor o menor en fecha. En el menú desplegable de selección de rama, se puede observar que aparece una selección por defecto con el mensaje de *Elige una rama* que dentro de la etiqueta `<select>` de HTML, está definida como una opción desactivada de la lista de opciones, no se puede seleccionar y tampoco enviar. Cuando se envía una de las ramas como opción de filtrado de trabajos, ésta se queda marcada en la página que genera la consulta, gracias al atributo `selected` de HTML, aunque puede marcarse otra selección y enviar esa opción.

Al lado de la barra de filtrado de ofertas por rama o especialización, nos encontramos con una barra de búsqueda para ofertas [Figura 4.11].



Figura 4.11: Barra para buscar oferta en inicio

En esta barra, por medio del atributo `placeholder` de HTML, vemos un breve mensaje que nos explica cómo debemos realizar la búsqueda. Para ejecutar una búsqueda con tan sólo introducir una palabra o frase, nos devolverá una tabla con todas aquellas ofertas en las que se ha hallado una coincidencia a esa entrada en todos los distintos campos de información que consta una oferta. Esta acción se consigue, de la misma manera antes, con una consulta a MySQL que se formula del siguiente modo:

```
$consulta_busqueda="select num_ident, titulo, rama,
tutor, deadline from ofertas where UPPER(titulo) LIKE "
$consulta_busqueda.= "'%".$parametro_busqueda."%'
or UPPER(rama) LIKE "
$consulta_busqueda.= "'%".$parametro_busqueda."%'
or UPPER(tutor) LIKE "
$consulta_busqueda.= "'%".$parametro_busqueda."%'";
```

El resultado de esta consulta devuelve una tabla equivalente a la de filtrado por especificación, donde se pueden ordenar las ofertas obtenidas por título, rama, tutor y fecha límite.

Debajo de estas dos tenemos la tabla general de ofertas actuales, organizadas por defecto de más actuales a menor [Figura 4.12].

Listado de ofertas actual
Número total de ofertas actuales: 7

Título	tema	autor	Fecha de inscripción
Experiencia de usuario en los sistemas de la UXX	UX	Laura Martínez	2013-06-30
Web semántica	Machine Learning	Laura Martínez	2013-06-30
Marketing Online	Marketing	Laura Martínez	2013-06-29
Desarrollo de una red social de TICs	Telecomunicaciones	Laura Martínez	2013-06-29
Desarrollo de una red social de TICs	Telecomunicaciones	Laura Martínez	2013-06-29
H5	H5	Laura Martínez	2013-06-30
Evaluador de documentos y comentarios	Software libre/Software analítico	Laura Martínez	2013-06-29

Figura 4.12: Tabla de ofertas en inicio

Esta tabla, al igual que las dos anteriores, pueden ordenarse por medio de los campos que definen cada columna. Tanto en esta tabla, como en las anteriores, el título de cada registro es un enlace que lleva a una página donde se muestra toda información guardada. Además, si alguna de estas ofertas ha recibido muchas visitas, muchos usuarios han mirado su contenido, se mostrará un icono de un fuego al lado del título. Este icono resalta a la vista de cualquier usuario que es una de las ofertas que más atención está teniendo [Figura 4.13].

 Web semántica	Machine Learning	Laura Martínez	2013-06-30
Marketing Online	Marketing	Laura Martínez	2013-06-29

Figura 4.13: Aviso de oferta muy visitada

Para calcular si una oferta está siendo muy visitada por los usuarios, lo que se hace es una consulta a MySQL un tanto compleja. El código de esta consulta es el siguiente:

```
$consulta_hot ="
        SELECT
tabla_visitas.visitas AS visitas,
tabla_visitas.num_ident_oferta AS num_ident_oferta,
percentil.percentil_90 AS percentil_90,
IF(visitas >= percentil_90, 1,0) AS hot
FROM
(
        SELECT
            COUNT(*) AS visitas,
            num_ident_oferta,
            1 as join_var
        FROM visitas_ofertas
        GROUP BY num_ident_oferta, join_var
) tabla_visitas
LEFT JOIN
```

50CAPÍTULO 4. DESARROLLO DE LA HERRAMIENTA PARA LA CONCESIÓN DE T.F.G.

```
(
SELECT
    SUBSTRING_INDEX(
        SUBSTRING_INDEX(
            GROUP_CONCAT(
                visitas_agrupadas.visitas
                ORDER BY visitas_agrupadas.visitas
                SEPARATOR ','
            )
            , ','
            , 90/100 * COUNT(*) + 1
        )
        , ','
        , -1
    ) AS percentil_90,
1 AS join_var
FROM
(
    SELECT
        COUNT(*) AS visitas,
        num_ident_oferta,
        ganador
    FROM visitas_ofertas
    LEFT JOIN ofertas
    ON visitas_ofertas.num_ident_oferta = ofertas.num_ident
    WHERE ganador IS NULL
    GROUP BY num_ident_oferta
) visitas_agrupadas
) percentil
ON tabla_visitas.join_var = percentil.join_var
WHERE percentil_90 IS NOT NULL
";
```

Básicamente, lo que hacemos es calcular el percentil de las visitas de cada oferta, y si este percentil supera el 90% se marca como *hot*. Para ello, existe una tabla llamada *visitas-ofertas* donde guardamos el número de identidad de la oferta y la fecha de visita, definida como un *TIMESTAMP*. Cada vez que un usuario entre en esa oferta se añadirá una entrada a esta tabla.

El percentil se define hallando en número total de las veces que se ha cargado una página de una oferta. En la tabla *visitas-ofertas* se consulta el número de registros que tiene esa oferta pasándole el número de identidad de dicha oferta y se calcula el porcentaje de ese valor. Comparamos este porcentaje calculado, con el obtenido del resto de ofertas y determinamos cuál está dentro de un percentil mayor o igual al 90 % [Figura 4.14].

(Solicitud enviada)

Título:	Web semantycs	 <p>¡Está que arde!</p> <p>Esta oferta es una de las ofertas más visitadas</p>
Rama:	Machine Learning	
Tutor:	Laura Martínez	
Email de contacto:	lauraborlaff@hotmail.com	
Departamento:	AI	
Descripción:	Ejemplo de oferta	
Conocimientos:		
Tecnologías:		
Fecha límite de inscripción:	2013-06-30	

(Última actualización: 2013-05-30 00:18:53)




Figura 4.14: Página de una oferta muy visitada

Para que la información en todas las tablas de nuestro proyecto quede representada mucho más clara para el usuario, hemos hecho que cuando el usuario pase el ratón por cada una de las filas se ilumine la fila entera de la tabla [Figura 4.15].

Título <small>Δ</small> <small>∇</small>	Rama <small>Δ</small> <small>∇</small>	Tutor <small>Δ</small> <small>∇</small>	Fecha límite inscripción <small>Δ</small> <small>∇</small>
Experiencia de usuario en los sistemas de la URJC	UX	Laura Martínez	2013-06-30
Web semantycs	Machine Learning	Laura Martínez	2013-06-30
Marketing Online	Marketing	Laura Martínez	2013-06-29
Desarrollo de una red social de TEGs	Telecomunicaciones	Laura Martínez	2013-06-29
Desarrollo de una red social de TEGs	Telecomunicaciones	Laura Martínez	2013-06-29

Figura 4.15: Iluminado de las filas debajo del puntero del ratón

Así conseguimos que el usuario esté mejor ubicación dentro de la página, aunque para entrar en la página donde se detalla cada oferta, se deberá hacer clic en el título. Sólo el título de las ofertas son enlace en estas tablas. Este efecto de resaltar las filas de la tabla se consigue mediante un código en HTML, en vez de CSS, como se podía esperar en un principio. Conseguirlo no fue tarea fácil, puesto que no había mucha información en la red sobre él. La mayoría de las páginas que trataban sobre este efecto, lo hacían por medio de CSS, pero tras varios intentos en nuestro código, nunca funcionó de la manera deseada o eran demasiado complejas.

Algunos procedimientos consistían en mezclar CSS3 con JavaScript [38]. Se definía un *script*, el cual contaba con una serie de funciones que hacían, que cuando se pusiera el ratón sobre una celda (la única que se conseguirá iluminar mediante CSS), se extendiera ese efecto al resto de la fila, cambiando el background de ese elemento fila. Fijando el mismo color del background para la fila, que el de la acción *hover* de HTML creada para la celda.

Otra solución es usar CSS3 y jQuery para resaltar las filas de una tabla [39]. En CSS se determinaba el selector *hover* para la fila como un objeto animado, más concretamente como una transición con un retardo y un efecto de salida o fin. Se utilizaba el código de JQuery para estimular este efecto de *hover* de una celda y luego se definía una clase para hacer el resaltado de la fila al completo, también implementando en ella el mismo efecto animado de la transición.

Pero nosotros, después de intentar todas estas formas, creíamos que se debía de poder hacer de una forma más sencilla y la encontramos [40]. En nuestras tablas, simplemente con la siguiente instrucción conseguimos ese efecto deseado:

```
echo '<tr onmouseover="this.style.backgroundColor=\'#f1f0f0\';"
onmouseout="this.style.backgroundColor=\'#ffffff\';">';
```

En la solución que proponemos, ejecutamos un JavaScript cuando el ratón se mueve dentro de un elemento, en este caso fila. Cuando el evento de ratón está acitvo sobre nuestro elemento, cambiamos el fondo de ese elemento al especificado en *onmouseover*, y una vez que se finalice volvemos a poner el anterior fondo del elemento con *onmouseout*. Estos parámetros están soportados por la mayoría de los navegadores (Internet Explorer, Mozilla Firefox, Opera, Chrome, Safari), por lo que no hay problema de compatibilidad.

Esta página de inicio se mantiene igual entre los roles de alumno, profesor y administrador, conserva la misma estructura. Pero en el rol de alumno se añade una funcionalidad más. El rol de alumno dispone de un elemento incorporado en HTML5, el *canvas*. Este canvas lo hemos realizado semejante a un cartel luminoso con una frase en movimiento [Figura 4.16].



Figura 4.16: Canvas en inicio alumno

En él se avisará a los usuarios si hay ofertas que estén a punto de expirar su fecha límite de inscripción. Definiendo este período de tiempo como hasta una semana respecto al día actual y la fecha tope. Este mensaje mostrará los títulos de aquellos TFG que vayan a cerrar inscripciones, y en caso de no haber ninguno, se observará el respectivo aviso informativo que no hay trabajos de fin de grado que tengan su fecha a punto de concluir. Para crear ese mensaje hemos realizado la siguiente consulta a MySQL:

```
$query="select titulo, datediff(deadline, date(now()))
as dias_restantes from ofertas where date_add(date(now()),
INTERVAL 7 DAY) >= deadline AND date(now()) < deadline
AND estado='No decidido'";
```

El resultado que genere esta consulta representará el mensaje a mostrar. Este mensaje se pasa a un código JavaScript para ejecutar su animación. El código de dicha animación está contenido en el fichero *led-scroller.js* y su respectivo CSS. Para realizar este código me he basado en un código de Ray Hammond [41]. Cuando se carga la página de inicio de alumno, se llamará a una función "*init()*" que se incluye dentro del fichero JavaScript. El primer paso que hace esta función es obtener un identificador para el elemento canvas, si existe podemos garantizar que el navegador soporta el canvas. El segundo paso es cargar las imágenes que contiene las letras del letrero luminoso en movimiento. Cuando se produzca el evento del canvas se activará la función *startAnim()*. En la función *startAnim()* se recupera el mensaje enviado por parámetro desde el fichero inicio, previamente convertido a texto en mayúsculas, ya que las imágenes que contienen las letras del letrero sólo tienen letras en mayúsculas. Esta función tiene las variables globales de *iDrawPhase* y *iStep* que se utilizarán en el bucle principal. Al mensaje enviado se le añaden espacios al principio a modo de prefijo para que la primera letra aparezca en el extremo derecha del letrero y se vaya desplazando hacia el lado izquierdo cuando se inicie la animación. En la función *draw()* es donde se consigue dibujar el mensaje en el letrero. Cada una de las letras se trata individualmente. El primer bucle itera sobre cada caracter. El bucle garantiza que la letra a representar está dentro de los rangos límite para que sea posible su impresión. Estos límites se establecen para que coincidan con las letras que tenemos disponibles en imagen. Disponemos de las letras de la A a la Z en mayúsculas. En este código de JavaScript utilizamos el código ASCII numérico para identificar el caracter actual está en ese rango. Para determinar la ubicación de cada letra dentro del canvas, necesitamos calcular una correlación entre las letras y cada una de las celdas en que dividimos el canvas:

```
iSpriteCol = Math.abs(65 - iCharCode);
iSpriteRow = 0;
```

Estos dos valores se pasan a la función *drawLetter*. En ella, la posición en la pantalla de la letra se obtiene, multiplicando la posición de la letra dentro del mensaje por el ancho del elemento canvas. Este se ajusta decrementando la variable *iStep* cada vez que el temporizador se ejecuta, lo que logra el efecto de la animación. Se realiza una comprobación para asegurar la posición 'x' calculando que está dentro de los límites de la pantalla. Si es así, se dibuja la letra en la posición obtenida. Después de escribir el mensaje completo, añadimos en el código unos espacios en blanco. Esto hace el efecto de trasladar el último caracter del mensaje totalmente a

la izquierda y hacerlo desaparecer. Una vez hecho esto, se comprueba que se ha escrito todo el mensaje entero y de ser así se detiene el bucle de ejecución del temporizador [41].

El último elemento que se encuentra en la página de inicio en los tres roles de usuarios es un menú desplegable para elegir el estilo que más guste al usuario. Cada vez que este usuario entre en su cuenta, se cargarán todas las tablas con este CSS elegido [Figura 4.17].



Figura 4.17: Barra para elegir el estilo de las páginas de usuario en THAIS

Página historial de TFGs

Esta página muestra una tabla con todas las ofertas que se han concedido en nuestra herramienta, con la misma estructura que la tabla de ofertas de la página de inicio, aunque se cambia la columna de fecha límite de solicitud por una en la que se refleja el nombre del alumno al que se le ha otorgado el trabajo. Es un historial de todas las ofertas otorgadas a alumnos. Esta página es igual para todos los roles de usuario. En ella, al igual que en inicio, los títulos de las ofertas son enlaces y la fila se resalta al pasar el ratón por encima, como explicamos en el apartado anterior. El usuario administrador es el único que puede borrar las ofertas directamente de la tabla [Figura 4.18].

Título	Rama	Tutor	Departamento	Alumno asignado
Sido	Sido	Laura Martínez	SIS	Laura Martínez Botani

Figura 4.18: Página de historial de TFG

El profesor, que sea el responsable de una oferta, también podrá eliminar su registro, pero una vez entrando en la página de la oferta. El motivo de que lo hayamos hecho así es porque quisimos darle al usuario administrador la responsabilidad de limpiar los registros antiguos cuando éste deseara. Al ser una página dedicada a presentar las antiguas ofertas publicadas y realizadas, tiene un carácter meramente informativo, por que lo que tampoco es muy necesario borrar antiguos registros. Cuantas más propuestas haya en esta lista, más orientado podrá estar el alumno que lo consulte sobre este tipo de trabajos. Por eso, lo dejamos en la mano del usuario administrador. Si se deseara eliminar los registros automáticamente, se podría hacer un script que borrara las ofertas pasado un tiempo específico sin necesidad de intervención. Propusimos algo así, en una de nuestras primeras versiones, pero al final, fue descartada, porque entonces estaríamos entrando en una cuestión de qué fecha sería la apropiada a partir de la cual eliminar los registros de las ofertas y ahí no todo el mundo coincide. Además, nunca es recomendable borrar registros, puesto que nunca se sabe cuando se necesitarán, almacenándolos en la base de

datos, siempre estarán a mano para posibles futuras consultas. De todas maneras, implementar una alternativa para que, una vez acumuladas muchas ofertas, sólo se impriman en éstas a partir de un tiempo dado, no sería nada difícil, simplemente cambiando un poco la consulta que hacemos a MySQL bastaría.

Veamos, la consulta que hemos dejado implementada es la siguiente:

```
$consulta="select ofertas.num_ident As num_ident,titulo,
rama,tutor,departamento,id_ganador, nombre,
apellidos from ofertas LEFT JOIN
aceptaciones ON ofertas.num_ident = aceptaciones.id_oferta
LEFT JOIN usuarios ON aceptaciones.id_ganador = usuarios.num_ident
where id_ganador IS NOT NULL
order by ultima_actualizacion asc";
```

Si queremos limitar las ofertas a imprimir por pantalla por aquéllas hasta una fecha dada, por ejemplo seis meses (180 días), tendremos que modificar esta consulta de una forma similar a la utilizada en el elemento de canvas en el inicio de alumno. Quedaría de la siguiente forma:

```
$consulta="select ofertas.num_ident As num_ident,titulo,rama,
tutor,departamento,id_ganador, nombre, apellidos,
datediff(deadline, date(now())) as
dias_restantes from ofertas LEFT JOIN aceptaciones ON
ofertas.num_ident = aceptaciones.id_oferta LEFT JOIN usuarios
ON aceptaciones.id_ganador = usuarios.num_ident where
id_ganador IS NOT NULL and
date_add(deadline, INTERVAL 180 DAY) > date(now()) AND
date(now()) > deadline order by ultima_actualizacion asc";
```

Tras esta consulta, MySQL sólo nos devolvería aquellas ofertas otorgadas cuya fecha límite sumada a 180 días, como tiempo de ventana de selección, sea mayor que la fecha actual. El resto del código implementado se mantendría intacto, puesto que se trabaja en función de la respuesta de MySQL y al haber preguntado por los mismos campos, no habría que modificar nada más.

Página de las ofertas

Un usuario llega a esta página, tras hacer clic sobre el enlace de título en cada una de las tablas que hemos comentado anteriormente o desde una URL a una de nuestras ofertas compartida en la red social Twitter. Desde este enlace, se pasa como parámetro en la URL el número de identidad de la oferta. Con este número podremos reconocer la oferta, en la nueva página y consultar a la base de datos por su información completa [Figura 4.19].



Figura 4.19: FPágina de una oferta en THAIS (Alumno)

La información al completa de la oferta aparece en la página organizada en dos columnas. Esta apariencia la conseguimos gracias a una tabla en HTML sin bordes. Las ofertas cuentan con los siguientes campos: título de la oferta, especificación de la oferta, nombre del tutor (con apellidos), correo electrónico de contacto con el tutor, departamento al que pertenece el trabajo, descripción del proyecto, conocimientos necesarios que debe tener el posible candidato, tecnologías que se utilizaran en el desarrollo del trabajo y fecha límite para solicitarla. Con todos estos campos consideramos que el trabajo presentado queda perfectamente definido, además, es un formato que ya se estaba utilizando dentro de la Escuela.

Si la oferta es una de las más visitadas aparecerá un mensaje a la izquierda de la página informando de ello, así como el icono de un fuego que también utilizamos en inicio. Cómo, lo explicamos en el apartado de la página de inicio. Cuando la oferta ya tiene un alumno asignado, aparecerá el nombre de éste y pasará a estar publicada a la página de trabajos ya concedidos y no se mostrará en inicio, a no ser que se deniege a este alumno la oferta.

Esta página tiene una ligeras diferencias dependiendo del rol de usuario que la consulte. Si se trata de un alumno, se mostrará la página hasta ahora descrita. Pero si se trata de un profesor, aparecerá una tabla con todos los nombres y los emails de los alumnos que la han solicitado. Los nombres de estos alumnos estarán enlazados a su perfil dentro de nuestra plataforma. Los alumnos no pueden ver esta tabla puesto que no queremos que rivalicen unos con otros, queremos que cuando se inscriban en una oferta sea libremente. Otra de las razones fue pensando en la privacidad del alumno. El que un alumno solicite una oferta no tiene que ser de interés para nadie más que del profesor que la ha creado y que escoge al candidato. Los alumnos pueden sentirse seguros en este aspecto, ningún otro alumno podrá ver sus candidaturas a ofertas. Además, así también evitamos que los alumnos curiosean los perfiles de sus compañeros, lo que creemos que proporciona más ventajas que desventajas. Como nadie más que los profesores (y el administrador) pueden ver el perfil de un alumno (a no ser que éstos se lo enseñen directamente a alguien), dejamos a relucir la capacidad del alumnos a redactar su perfil de la forma más atractiva posible que pueda y que, por supuesto, será valorada por el tutor de la oferta en su decisión. Con esta simple acción, creemos que motivamos al alumno a ser creativo, conciso

y claro a la hora de presentar sus datos. Esto, sin duda, le servirá de mucha ayuda, una vez acabados sus estudios, a la hora de escribir su currículum y buscar un puesto de trabajo en una empresa. Esto es algo que no se enseña en las asignaturas de la Escuela, y nos referimos a asignaturas troncales, obligatorias y optativas. Si un alumno desea información sobre cómo hacer un buen currículum puede realizar asignaturas ofertadas de libre elección o talleres de carácter temporal, que a veces, la Escuela organiza. Pero en estas opciones, en ninguna, se asegura llegar a la totalidad de los alumnos y desde nuestra plataforma, en cambio, sí podemos asegurar eso.

El tutor de la oferta y el administrador del sitio, además contarán con un botón para realizar la acción de seleccionar al alumno que considere más capacitado. Tras elegir a un candidato, desaparecerán los botones para seleccionar candidato y aparecerá un botón para deshacer esta elección en la fila del estudiante elegido. Igualmente, tutor y administrador dispondrán de las funciones de edición de la información de la oferta y eliminar la oferta. Hemos dado estos poderes también al administrador, como mecanismo de protección por si existen errores, que el servicio de la herramienta no se vea parado mientras se espera una solución. Siempre contamos con una actitud ética por parte del administrador, como un usuario que vela por el buen funcionamiento de la plataforma y sin ser el motivo de problemas secundarios.

Al final de esta página se encuentran los botones para hacer un tweet de la oferta en tu perfil de Twitter y hacer +1 en la red social GooglePlus para lo usuarios puedan compartir en sus redes las ofertas que más interesantes les ha podido parecer y notificárselo a sus conocidos.

Página de perfil de usuario

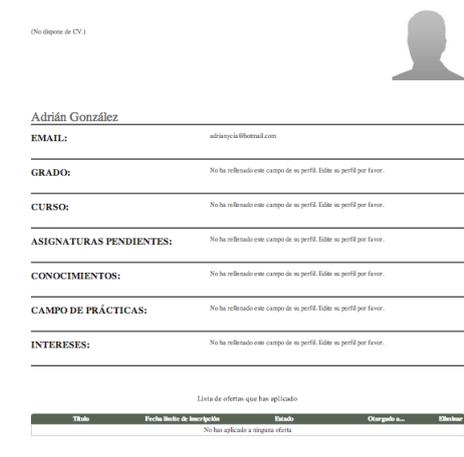
Los usuarios con rol de alumno tienen un perfil donde pueden mostrar datos personales. Los campos que hemos escogido para definir este perfil intentan aportar toda la información que pueda necesitar un profesor para designarle el trabajo que ha solicitado o no. Estos campos son: nombre, apellidos, correo electrónico de contacto, grado que curse, curso matriculado al que asiste, asignaturas pendientes de otros años, conocimientos que desea resaltar, campo de las prácticas que realice e intereses. En el campo de interés se trata de reflejar aquello por lo que el alumno se muestre interesado, ya sea una especificación de la carrera, alguna tecnología específica, algún lenguaje de programación, algún tema que crea relevante, etc. El alumno es libre de rellenar cada campo a su gusto, pero nunca deberá de olvidarse, que esta herramienta está destinada hacia la educación profesional y no para el ocio, por lo que la forma en que lo redacte puede aventajarle o descalificarle automáticamente de un trabajo que solicite. Debe tener conciencia para publicar sus datos de forma profesional y clara. Aparte, es muy recomendable que lo haga de la forma más precisa, puesto que así los profesores podrán deliberar en qué trabajo encaja mejor. Aunque un alumno tenga que exponer las asignaturas que todavía no ha superado y esto pueda hacerle sentir que puede estar en desventaja en las ofertas a las que aspire, de ningún modo esto es así. De esta forma el profesor tiene constancia de los puntos

fuertes del alumno y los que no son tan fuertes. Sólo recercurtiría negativamente en la decisión, si se necesitara un grado determinado de conocimiento en alguna oferta y que éste fuera garantizado con la superación de una materia. Por otro lado, como profesores y alumnos se conocen entre sí, si el profesor de esa materia conoce el rendimiento académico de ese alumno y cree que es adecuado, este factor, sin duda, quedaría relevado a un segundo plano.

El alumno puede publicar sus redes sociales: Twitter, GooglePlus o LinkedIn. Para ello, deberá de añadir esos datos en la página para editar perfil, dentro del menú de 'Perfil'. Estas redes aparecerán en la parte de arriba, a la derecha de la página. Esto puede ser de utilidad para los profesores, sobre todo LinkedIn, que contiene campos más orientados hacia lo profesional que educativo. Añaden más información sobre el candidato sin la necesidad de que éste tenga que rellenar los mismos campos otra vez en el perfil de nuestra plataforma. Hoy en día con la extensión del las redes sociales, cada persona cuenta con gran variedad de cuentas y en casi todas ellas debemos rellenar los mismos campos de perfil, lo que resulta bastante cansado, y si encima, nosotros los requiriéramos otra vez irritaríamos al usuario. Por eso, tuvimos en cuenta esto, y preguntamos distintos datos, sabiendo que existía la posibilidad de adquirir datos del perfil más profesional del usuario por otras redes.

Igualmente, si el usuario lo desea, puede adjuntar su currículum vitae y aparecerá como un enlace situado al principio de la página donde se podrá descargar. Para ello, el alumno deberá subir dicho documento en el formulario de editar perfil y se guardará en la base de datos cvs.sql. En esa base de datos almacenamos el documento junto con el numero de identidad del alumno, el formato del documento, el tamaño, el nombre del archivo y le damos un número para identificarlo. Una vez guardado, la próxima vez que el alumno o un profesor entre en su perfil verá en la parte superior de la página un enlace de descarga directa. Este enlace llama a un script de PHP. En él se hace una consulta a MySQL para obtener el registro que contiene el CV del alumno que lleva esa identidad.

El usuario, también, puede adjuntar una foto de perfil. Si no sube ninguna foto, se mostrará una por defecto en su perfil. La forma de proceder para la foto es análoga a la explicada en el currículum vitae [Figura 4.20].



(No dispone de CV)

Adrián González

EMAIL: adriancia@hotmail.com

GRADO: No ha rellenado este campo de su perfil. Edite su perfil por favor.

CURSO: No ha rellenado este campo de su perfil. Edite su perfil por favor.

ASIGNATURAS PENDIENTES: No ha rellenado este campo de su perfil. Edite su perfil por favor.

CONOCIMIENTOS: No ha rellenado este campo de su perfil. Edite su perfil por favor.

CAMPO DE PRÁCTICAS: No ha rellenado este campo de su perfil. Edite su perfil por favor.

INTERESES: No ha rellenado este campo de su perfil. Edite su perfil por favor.

Lista de ofertas que has aplicado

Título	Fecha límite de inscripción	Detalle	Organismo a...	Estado
No has aplicado a ninguna oferta.				

Figura 4.20: Perfil de alumno en THAIS

La foto adjuntada en el formulario de edición de perfil se guardará en una base de datos además de su tamaño, el nombre del archivo, el formato, la identidad del usuario y se le otorgará un número para identificarla que se guardará en el registro de la base de datos de los usuarios. Cuando un usuario entre en el perfil se consultará a la base de datos si el dueño del perfil subió una foto y en ese caso, se llamará a un script de PHP para proceder a su visualización. En caso de que no exista, se imprimirá la foto de usuario por defecto que dispone la herramienta.

Hemos querido que el aspecto del perfil para mostrar la información del usuario sea la similar a la de un currículum vitae. Queremos que cuando un profesor vea la página del alumno sienta que está en un entorno profesional más que estudiantil. Para este diseño nos hemos fijado en la estructura de varias currícula en la red. Hemos dividido los datos en dos columnas, como en las ofertas, donde una columna contiene en nombre de cada campo y la otra la información del guardada en ese campo. La forma de proceder a crear esas dos columnas es diferente a la realizada con las ofertas. Si recordamos, en las ofertas conseguimos ese estilo mediante una tabla en HTML sin bordes. En cambio, aquí, logramos ese efecto por CSS. No procedimos de la misma manera, puesto que en esta página separamos cada campo con unas líneas horizontales que le dan ese aspecto de currículum, y con una tabla no quedaban del todo bien situadas. Por eso, recurrimos al CSS para alcanzar ese estilo. Para ello, dividimos la información que queremos en cada columna por medio de las etiquetas `<div>` en HTML y les damos las identidades de `'col1'` y `'col2'`. Una vez en el CSS, básicamente, lo que hacemos es dar a cada columna un ancho y una posición dentro de la página. A la primera columna, la que contiene los nombres de los campos, la damos una posición absoluta dentro del CSS y un ancho de 400 píxeles. A la segunda columna le ponemos un margen por la izquierda de 420 píxeles (para que no colisione con la anterior) y un margen por la derecha de 20 píxeles para que quede centrado. Además, la división del texto por columnas está contenida dentro de otra división que se encarga de dar al

texto en las columnas el estilo deseado como el tamaño de la letra, alineación, la indentación, etc.

De la misma forma que en la página de oferta, ésta tampoco es igual dependiendo del usuario que la consulte. Cuando un profesor o el administrador visita el perfil de un usuario sólo ve la información detallada hasta este punto. Por el contrario, si quien visita la página es el usuario del perfil podrá advertir una tabla en la parte final de la página. En esta tabla se muestran las candidaturas que el usuario está esperando su resolución. Estos registros no los puede borrar puesto que creemos que resultaría interesante que el estudiante pudiera ver su historial personal de solicitudes mientras no haya aceptado un TFG. Hacemos esta tabla privada para que no influya en la decisión del profesor sus anteriores candidaturas, tanto para bien como para mal. Donde para bien nos referimos a alumnos que hayan optado por trabajos más exigentes y que pongan en relieve la capacidad de compromiso del alumno, y, para mal, si le ha sido denegada una oferta similar anteriormente, no siendo el motivo que influyan en las nuevas solicitudes. Consideramos que esto es algo que entra dentro de la privacidad del alumno. Desde esta tabla se puede llegar a la página de las ofertas que estén presentes, puesto que están enlazadas para que el usuario no tenga que estar buscando la oferta en la página de inicio para consultar su información. Puede ir directamente desde su perfil y seleccionar la oferta entre sus candidaturas. En esta tabla además se le informará del estado de la oferta y a quién se le ha otorgado. Una vez que se haya elegido a un alumno, el alumno no podrá eliminar ese registro, se quedará guardado en la base de datos de candidatos.sql y será mostrado en su perfil. Mientras no haya un alumno seleccionado para la oferta, el usuario puede derogar esa candidatura, ya sea porque haya aplicado por error o cambio de idea hacia ella. En el caso de que un alumno sea elegido para una oferta y haya cambiado de idea sin que le diera tiempo a revocar su solicitud, no pasará nada puesto que hasta que no acepte dicha oferta no será oficial. Cuando a un alumno se le asigne la oferta que desee, podrá aceptar esa asignación directamente desde su tabla de solicitudes. Lo mismo ocurriría si un alumno es seleccionado para dos trabajos. Cuando un alumno acepte una oferta que le han asignado, desaparecerá la tabla de candidaturas en su perfil y sólo se mostrará la tabla con la oferta de trabajo de fin de grado que se ha comprometido a realizar.

Página de editar perfil

Esta página contiene un formulario con campos editables para la información del perfil del alumno. Si un alumno ya había almacenado algún dato, éste se mostrará en el campo determinado y en caso contrario, se mostrará vacío. En un campo vacío aparecerá un mensaje por defecto para explicar el contenido que se debería aportar mediante el nuevo argumento de HTML5 *placeholder*, y tendrá un estilo dentro del CSS. Al hacer el registro, el alumno ya cuenta con los campos: nombre, apellidos y correo (y esos campos aparecerán rellenos en esta página automáticamente), pero puede rellenar también los siguientes: el grado que estudia, el curso,

las asignaturas que tiene pendientes de otros cursos, los conocimientos que quiere destacar, el campo de prácticas, los intereses o lo que quiere aprender, la cuenta de Twitter, GooglePlus, LinkedIn, una foto para el perfil y currículum vitae [Figura 4.21].

The image shows a web form titled "Formulario editar perfil". It contains the following sections and fields:

- Nombre*:** Input field for first name.
- Apellidos*:** Input field for last name.
- Cursos:** Input field for course name.
- Correo electrónico*:** Input field for email address.
- Grado:** Input field for degree.
- Grado que cursas:** Input field for current degree.
- Cursos:** Radio buttons for selecting a course level: 1º, 2º, 3º, 4º.
- Asignaturas pendientes:** Text area for listing pending subjects.
- Conocimientos:** Text area for listing knowledge.
- Campo de prácticas:** Input field for listing practices.
- ¿Qué quieres aprender?:** Text area for listing interests.
- Twitter:** Input field for Twitter username.
- Google+:** Input field for Google+ profile URL.
- LinkedIn:** Input field for LinkedIn profile URL.
- Foto de perfil:** File upload field for profile picture.
- Currículum vitae:** File upload field for CV.

An "Actualizar" button is located at the bottom right of the form.

Figura 4.21: Formulario editar perfil en THAIS

En los campos de grado, asignaturas pendientes, conocimientos e intereses el alumno deberá redactar estos campos. En la información sobre las asignaturas no superadas, lo hicimos de esta manera y no mediante un listado donde el alumno pudiera seleccionar estas materias, porque creímos que resultaría muy incómodo para el alumno, tener que buscar cada una de las asignaturas, de cualquiera de los cursos. Como también contamos que los alumnos que hagan un uso más activo de esta plataforma sean alumnos de últimos cursos, es razonable pensar que no tendrán muchas asignaturas pendientes. Por lo que escribir el nombre de cada una no resulta del todo engorroso. En cambio buscar cada una de esas asignaturas o asignatura en singular en una lista/desplegable que comprenda todas las materias de todos los cursos sí que puede resultar molesto.

Como comentamos en el modelo de M.V.C., cuando un alumno envía el formulario, el primer paso es validar su contenido por medio de la clase `validadorFormularios.php`. El modo de comprobar que los datos son "correctos" depende del campo en cuestión. En todos se comprueba la longitud, restringida por cómo definimos estos datos en la base de datos para optimizar el espacio en la memoria. Pero en los campos obligatorios para formar un usuario (nombre, apellidos u correo) también comprobamos que el dato introducido sólo contenga letras para nombres y apellidos, ya que no existe nadie que tenga un dígito en ellos, y en el correo que pueda conte-

ner letras, símbolos, números, etc., además de comprobar el formato de una dirección de correo electrónico, es decir, debe tener un caracter @ y un dominio. El código para esta validación es:

```
function validarEmail ($email){
    echo '<script> validarEmail("' . $email . '") </script>';
    return (preg_match('/^[a-zA-Z]+([\.]?[a-zA-Z0-9_-]+)*
    @[a-z0-9]+([\.-]+[a-z0-9]+)*
    \.[a-z]{2,240}$/' , $email));
}
```

Si alguno de estos campos da error, se devolverá un mensaje de error y los campos del formulario rellenos con la información previamente enviada. Si todos los campos son adecuados, el programa llamará a la clase GestorUsuarios.php, que hace que se actualicen o añadan. Antes de actualizar la información, si el usuario, por ejemplo, cambia su dirección de correo electrónico, la que se utiliza para entrar en nuestra plataforma y debe ser única en nuestro sistema, se comprueba si ya existe un usuario que está usando de esa dirección. En caso afirmativo, se devuelve el mensaje de error y los campos como hemos comentado antes. En caso negativo, al enviar ese paquete con la nueva información al gestor también incluimos el número de identidad de éste, por lo que cuando actualizamos los datos, la forma de identificar al usuario a actualizar es por medio de ese número, y así evitamos errores de actualizar el usuario equivocado. Obviamente, descartamos la posibilidad de hacerlo mediante el nombre y apellidos puesto que puede haber alumnos con nombres parecidos o iguales, y lo mismo con los apellidos. Y tampoco era buena idea identificar al usuario a actualizar con el correo electrónico, puesto que un usuario puede ser libre de cambiar este campo a la cuenta que prefiera, y podría provocar ambigüedades. Lo mejor era utilizar un dato que identificara al alumno y fuera constante siempre. Sólo cuando se verifica la identidad del alumno y se comprueba que los datos son los adecuados se actualizará la información del perfil.

En los campos de la red social de Twitter el usuario deberá introducir su nombre de usuario. En el de GooglePlus y LinkedIn su URL. Para que funcionara bien el botón para añadir a los círculos dentro de la red de Google, hemos tenido que parsear la URL que introduce el usuario para utilizar sólo el dato que realmente identifica al usuario, y mantener el resto de la URL constante. Por lo que la URL que se sincroniza con la red ya está introducida en nuestro código, y cuando un usuario introduce la suya, simplemente tomamos el trozo que nos interesa para formar la URL completa. Tuvimos que tomar esta medida, porque el código que genera GooglePlus para su botón, el código que agregamos al nuestro para que funcione este botón, viene con una URL predefinida que difiere un poco, pero lo suficiente, de la dirección a la página del perfil; no son iguales. En el campo de la red LinkedIn el usuario deberá introducir la URL a su perfil público. Para los campos de redes sociales, simplemente nos limitamos a actualizar el

dato introducido en la base de datos, puesto que no tenemos forma de comprobar que el dato introducido pertenece al alumno o es el correcto, ya que no podemos adivinar el nombre de usuario en Twitter, la URL de su perfil en GooglePlus o LinkedIn.

Para el currículum y la foto, seguimos un procedimiento parecido al anterior. Creamos un array con los siguientes datos: nombre del fichero, la foto, fichero temporal, tamaño de la foto, tipo de la foto (su extensión: .jpg, .png o .jpeg) y número de identidad del alumno, en el caso de la foto. En el caso de un CV, el array es: nombre del fichero, fichero temporal, fichero adjuntado, tamaño, extensión y número de identidad del alumno. Antes de insertar estos datos en la base de datos, hacemos una comprobación sobre el tamaño del archivo subido. Hemos puesto como límite de 2 MBytes aproximadamente. Sólo si esa validación devuelve un valor verdadero, se procede a la actualización de la base de datos. Cuando se actualiza, primero se comprueba si ya había subido una foto antes, para elegir la consulta a la base de datos que proceda. Si ya existía una se hace una consulta previa para consultar la identidad de la foto en la base de datos que guarda los registros de fotos de todos los usuarios y se hace una actualización de los registros con esta ID, y, si no existía ninguna antes, se agrega ésta. Para el currículum el procedimiento es el mismo.

Si todo el proceso de edición de perfil ha funcionado como se esperaba, se redirigirá automáticamente al perfil del alumno, donde podrá apreciar los nuevos cambios.

Página de cambio de contraseña

Esta página se encuentra en los sitios de los usuarios de cualquier rol: alumno, profesor y administrador. Se trata de un formulario para cambiar la contraseña del usuario en cuestión en nuestra plataforma [Figura 4.22].

El formulario tiene un título "Cambiar password" en azul. Contiene dos campos de texto con etiquetas "Contraseña*" y "Repita la contraseña*", cada uno con un subtexto "Escriba la nueva contraseña" y "Repita la nueva contraseña" respectivamente. En la parte inferior derecha hay un botón azul con el texto "Cambiar".

Figura 4.22: Formulario cambio contraseña en THAIS

El usuario deberá introducir la nueva contraseña dos veces para su comprobación. Al pulsar el botón de 'Cambiar' sobre el formulario, enviamos las dos contraseñas a GestorUsuarios.php que a su vez envía una consulta a ValidadorFormularios.php. En ValidadorFormularios.php comprobamos que las contraseñas son iguales y su longitud es mayor que cuatro caracteres. Si devuelve un valor verdadero esta consulta, GestorUsuarios.php remitirá esta respuesta a cambiar_password.php y este script enviará la petición del cambio al gestor, en un paquete con

el correo del usuario, su identidad y la contraseña introducida. Y se procederá a realizar una actualización de este campo en la base de datos al usuario con esos datos.

Página de cambio de contraseña por petición al administrador

Cuando un alumno o profesor pida al administrador resetear su contraseña porque se le haya olvidado, hemos creado este formulario. El usuario se pondrá en contacto con el administrador del sitio para pedirle el cambio. El administrador dispone dentro de la opción de Usuarios de su menú, una opción dentro de este submenú para el cambio de contraseña de los usuarios. Como ya hemos dicho, siempre suponemos una buena conducta del administrador dentro de nuestra aplicación. Dentro de ese formulario el administrador sólo deberá introducir el correo del usuario que use en nuestra plataforma [Figura 4.23].



Figura 4.23: Formulario cambio de contraseña por petición al administrador

Al enviar este correo, se consulta la identidad de este usuario que desea el cambio para indicarlo en la actualización. Se originará un número aleatorio de cinco cifras que será la contraseña temporal del usuario. Esta contraseña se introduce en su registro de usuario de nuestra base de datos y se envía un correo automático al usuario informándole del cambio y cuál es su contraseña temporal. Además se le pide que entre lo antes posible en su sesión y vuelva a cambiar esta contraseña por una de su gusto.

Página de cambio de rol de usuarios

Esta página únicamente existe para el administrador. Es bastante parecida a la de cambio de contraseña de usuarios por medio del administrador. Tenemos un formulario con dos campos [Figura 4.24].

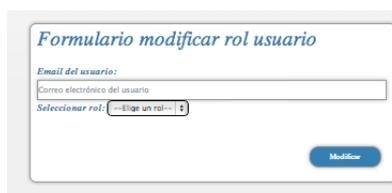


Figura 4.24: Formulario cambio rol

En el primero se deberá introducir la dirección de correo electrónico que esté usando el usuario en nuestra herramienta. La segunda es un menú desplegable donde elegir el nuevo rol: alumno, profesor o administrador. Tenemos que tener en cuenta que todos los nuevos usuarios se crean como alumnos por defecto. El usuario administrador se crea automáticamente al crear la base de datos en la máquina virtual. Este cambio de rol se efectúa inmediatamente tras el envío de este formulario completo. Este dato se envía al gestor que es el que realiza la consulta de actualización en la base de datos de usuarios. Tras el cambio de rol, se enviará un correo al usuario que se le han cambiado los permisos un correo avisándole del cambio e indicándole el nuevo rol.

Página para eliminar usuarios

La página para eliminar usuarios sólo está disponible en la sesión de administrador dentro de su menú Usuarios. En ella, el administrador introducirá el correo electrónico del usuario a borrar en un formulario [Figura 4.25].

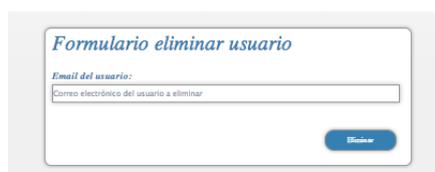


Figura 4.25: Formulario para eliminar usuario

Al enviar este dato se eliminará en ese mismo momento el registro del usuario en nuestra base de datos. Si la modificación se ha completado correctamente o se ha producido un error, se informará mediante un aviso en pantalla al administrador. Si ha finalizado con éxito se enviará un correo electrónico al usuario en cuestión notificándole la eliminación de su cuenta en nuestra herramienta.

Página para crear una nueva oferta

Tanto un profesor como el administrador pueden crear nuevas ofertas dentro de nuestra aplicación. Dimos esta acción al administrador en el caso de que un profesor tuviera un problema con esta acción en su sesión. Recordamos que el profesor que sea el tutor responsable de la oferta será el único (a excepción del administrador) que podrá editar su información más adelante, elegir al candidato o borrarla. Por lo que esto deberá de tenerse muy en cuenta a la hora de crearla. Si dos profesores cooperan en el desarrollo de una oferta, deberán elegir a uno de ellos como responsable de manejar la oferta en nuestra herramienta. Los campos de este formulario son los siguientes: título de la oferta, rama o especificación de la carrera, nombre completo del tutor responsable, email de contacto del tutor, URL a su página personal que contiene su perfil

dentro de la Escuela, departamento, descripción del trabajo, conocimientos requeridos para realizar la oferta, tecnologías que se usarán en el desarrollo y fecha límite para apuntarse [Figura 4.26].

Figura 4.26: Formulario para la creación de una nueva oferta

La forma de proceder para crear la oferta sigue la misma filosofía que en la creación de usuarios. Creamos un paquete con todos los nuevos parámetros y lo enviamos a un `GestorOfertas.php`. Este gestor lo reenvía a su vez a `ValidadorOfertas.php` y comprueba que los campos son válidos, revisa que la longitud del campo no excede el límite impuesto. En aquellos campos que marcados como obligatorios (título, tutor, email, departamento y descripción) se comprueba que se ha introducido un dato con una longitud mayor que cero. Cuando el validador devuelve una respuesta afirmativa a esta consulta, el gestor reenvía esta respuesta al formulario, y se envía la petición al gestor para insertar la nueva oferta en la base de datos. Cuando se inserta esta oferta en la base de datos se le otorga su número de identidad único en nuestra herramienta. Tras guardar el trabajo, se envía un *tweet* automático a la cuenta en la red social Twitter THAIS URJC informando a sus seguidores sobre la nueva oferta. En ese *tweet*, además incluimos la URL a la página de la oferta. Si todos los pasos del proceso han ido bien, se redireccionará al usuario a su página de inicio.

Cabe destacar, que cuando un usuario introduce la URL de su página personal en el respectivo campo del formulario, esto hace que en la página de la oferta ya creada, el nombre del tutor queda resaltado como enlace, donde si se hace un evento con el ratón abrirá dicha página personal. Esto lo hicimos así, como ya hemos comentado, porque los profesores no disponen de perfil en nuestra aplicación, por comodidad para éstos. Los profesores de la Escuela ya disponen de varios perfiles en las webs de la Universidad, ya sea del departamento, Moodle o la propia

Universidad, y consideramos que añadirles otro perfil más donde rellenar con los mismos datos sería bastante molesto para ellos.

Página para editar la información de la oferta

El formulario para editar la información contenida en una oferta se puede llegar a él por medio de la sesión del usuario administrador y el tutor responsable de la oferta. Si un profesor no es el tutor de una oferta no podrá editarla ni borrarla. Podrá ver la información de la oferta, la tabla de candidatos que la han solicitado y sus perfiles. Este formulario es el mismo que para crear una nueva oferta. Con la diferencia que cuando cargamos esta página, los datos previamente introducidos aparecen en sus respectivos campos para que se puedan modificar. Los campos que no hayan sido a priori rellenados, aparecerán vacíos con un *placeholder* como mensaje aclaratorio sobre el contenido que se pide. El procedimiento es el mismo que en la creación de la oferta, pero con la única diferencia es que tras pasar la comprobación del validador, cuando se envía la petición al gesto para que éste la actualice, enviamos el número de identidad de la oferta, para que no haya error y se modifique la oferta correcta.

Página de notificaciones

Esta página está disponible en las sesiones de profesor y alumno. No la añadimos al usuario administrador, porque no tendría mucho sentido. El administrador del sitio sólo debería ocuparse del buen funcionamiento del sitio y para ello no son necesarias las notificaciones. Además, si pudiéramos una página de notificaciones para él, sería tan larga que perdería cualquier utilidad. Tampoco las notificaciones que manejamos serían de su interés. Las notificaciones que mostramos son distintas dependiendo de cada rol.

■ Alumno

En el caso del alumno, verá noticias sobre las nuevas ofertas publicadas y el estado de las ofertas que ha pedido, si se le ha asignado la oferta [Figura 4.27].



Figura 4.27: Notificaciones alumno

En su perfil o en la página TFG concedidos podrá ver a quién se le ha otorgado. Para mostrar estas notificaciones hacemos una consulta a la base de datos un poco más sencilla que para los profesores:

68CAPÍTULO 4. DESARROLLO DE LA HERRAMIENTA PARA LA CONCESIÓN DE T.F.G.

```
$consulta="SELECT
    DATE_FORMAT(fecha, '%d-%m-%Y') AS fecha_notificacion,
    tipo_notificacion,
    id_oferta,
    titulo
FROM
(
    SELECT
        ultima_actualizacion AS fecha,
        'nueva_oferta' AS tipo_notificacion,
        ofertas.num_ident AS id_oferta,
        titulo
    FROM ofertas

    UNION ALL

    SELECT
        fecha_asignacion AS fecha,
        'oferta_asignada' AS tipo_notificacion,
        ofertas.num_ident AS id_oferta,
        titulo
    FROM ofertas
    JOIN asignaciones
        ON ofertas.num_ident = asignaciones.id_oferta
    WHERE
        id_ganador = ";
$consulta .= $num_ident_usuario;
$consulta .= ") AS notificaciones
ORDER BY fecha_notificacion ASC";
```

Con esta consulta directamente definimos el tipo de notificación para luego mostrar un mensaje u otro. Creamos dos tipos de notificación: nueva_oferta y oferta_asignada. Estos tipos de mensaje junto con la fecha de notificación, creada a partir de su fecha de creación guardada en nuestras bases de datos cuando se creó el registro oportuno. Hacemos una doble consulta, una a la base de datos de oferta para publicar las ofertas más recientes, y otra a la base de datos de ofertas y asignaciones para ver qué ofertas están asignadas.

■ Profesor

Para el rol de profesor se muestran notificaciones sobre los candidatos de sus ofertas. Si un candidato se inscribe en uno de sus trabajos, se mostrará un mensaje en esta página con la hora en la que tuvo lugar. También mostramos noticias acerca de si uno de sus candidatos asignados ha aceptado otra oferta que no era la del profesor que este ejecutando su sesión [Figura 4.28].



Figura 4.28: Notificaciones profesor

La consulta que hacemos a MySQL para mostrar la información para las notificaciones de los profesores es un poco compleja, se hace una triple consulta, como se puede observar en el siguiente código:

```
$consulta=" SELECT
    DATE_FORMAT(fecha, '%d-%m-%Y') As fecha_notificacion,
    tipo_notificacion,
    id_oferta,
    titulo,
    id_alumno,
nombre,
    apellidos
FROM
(
    SELECT
        fecha_candidatura AS fecha,
        'candidatura' AS tipo_notificacion,
ofertas.num_ident AS id_oferta,
        titulo,
        num_ident_candidato AS id_alumno,
        nombre,
        apellidos
FROM candidatos
JOIN ofertas
ON ofertas.num_ident = candidatos.num_ident_oferta
```

70CAPÍTULO 4. DESARROLLO DE LA HERRAMIENTA PARA LA CONCESIÓN DE T.F.G.

```
JOIN usuarios
ON usuarios.num_ident = candidatos.num_ident_candidato
WHERE
email_tutor = '"';
$consulta .= $username;
```

```
$consulta .= "'
```

```
UNION ALL
```

```
SELECT
aceptaciones.fecha AS fecha,
'oferta_propia_aceptada' AS tipo_notificacion,
ofertas.num_ident AS id_oferta,
titulo,
id_ganador AS id_alumno,
nombre,
apellidos
FROM ofertas
JOIN aceptaciones
ON aceptaciones.id_oferta = ofertas.num_ident
JOIN usuarios
ON aceptaciones.id_ganador = usuarios.num_ident
WHERE
email_tutor = '"';
$consulta .= $username;
```

```
$consulta .= "'
```

```
UNION ALL
```

```
SELECT
aceptaciones.fecha AS fecha,
'oferta_alternativa_aceptada' AS tipo_notificacion,
ofertas.num_ident AS id_oferta,
titulo,
aceptaciones.id_ganador AS id_alumno,
```

```

nombre,
apellidos
FROM asignaciones
JOIN ofertas
  ON asignaciones.id_oferta = ofertas.num_ident
JOIN aceptaciones
  ON asignaciones.id_ganador = aceptaciones.id_ganador
JOIN usuarios ON aceptaciones.id_ganador = usuarios.num_ident
WHERE
email_tutor = '';
$query .= $username;

$query .= "' AND aceptaciones.id_oferta NOT IN
(SELECT num_ident FROM ofertas WHERE email_tutor = '";
$query .= $username;

$query .= "' )
) AS notificaciones
ORDER BY 1 ASC";

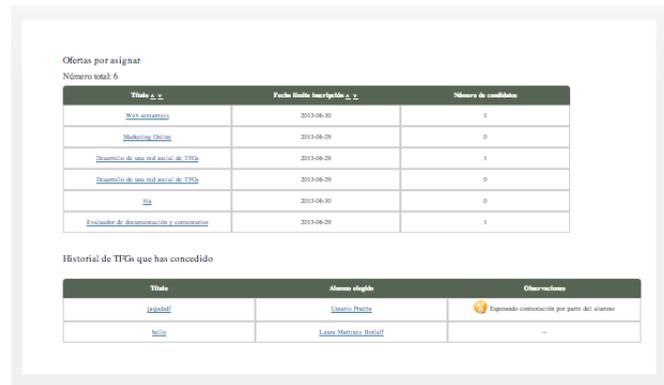
```

Esta consulta sigue el esquema de la realizada para las notificaciones de alumnos. Los tipos de notificaciones en este perfil son: *candidatura*, *oferta_propia_aceptada* y *oferta_alternativa_aceptada*. Para el primer tipo de notificación hacemos una consulta de los campos necesarios a la tabla de candidatos. Para la consulta de *oferta_propia_aceptada* hacemos una consulta a la base de datos de oferta para sacar la información relevante de nuestras ofertas en base a esos datos juntamos y comparamos esos datos con las tablas de aceptaciones y usuarios para comprobar si uno de nuestros candidatos nos ha aceptado. El último tipo es similar a la consulta anterior, se compara la información de nuestras asignaciones con las de las aceptaciones para confirmar que el alumno seleccionado ha aceptado nuestra oferta o no.

Página de mis ofertas para profesor

Esta página sólo está disponible para profesores. En ella los profesores pueden hacer el seguimiento de sus ofertas publicadas. En esta página se muestra una tabla de las ofertas que aún están por asignar, donde, además, se puede ver el número de candidatos que tiene actualmente. El título de las ofertas está enlazado a su página donde podrán ver quiénes son sus candidatos

y sus respectivos perfiles. Asimismo, dentro de la página de la oferta podrán seleccionar al candidato o deshacer esta elección. También, en esta página de mis ofertas se dispone de una tabla con las ofertas que se han asignado [Figura 4.29].



Ofertas por asignar
Número total: 6

Título a. v.	Fecha límite inscripción a. v.	Número de candidatos
Web semestral	2015-06-30	1
Marketing Online	2015-06-29	0
Desarrollo de una red social de TFGs	2015-06-29	1
Desarrollo de una red social de TFGs	2015-06-29	0
IIS	2015-06-30	0
Evolution de documentación y comentarios	2015-06-29	1

Historial de TFGs que has concedido

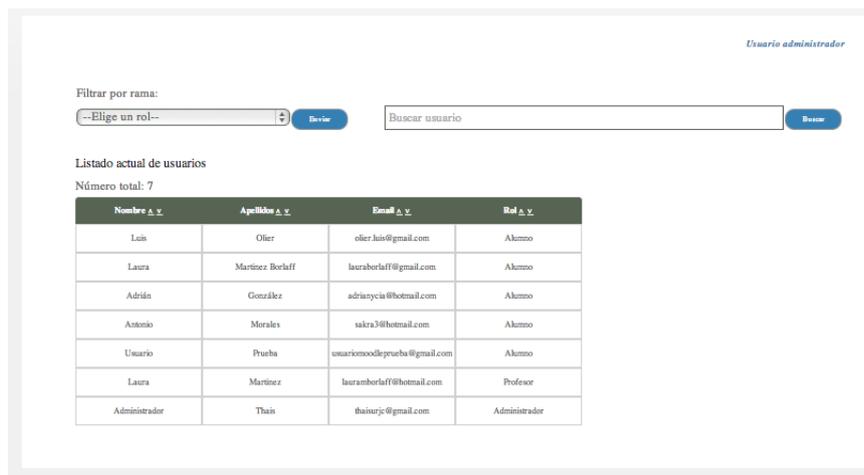
Título	Alumno asignado	Observaciones
IIS	Luisa Prieto	Esperando confirmación por parte del alumno
IIS	Luisa Martínez Barja	-

Figura 4.29: Página con las ofertas del profesor usuario

Cuando un profesor asigna una oferta, esta oferta pasa de la tabla de ofertas por asignar a la de historial de ofertas concedidas. Si el alumno no ha aceptado todavía la oferta, aparecerá un mensaje en la columna de observaciones informando al profesor que se está esperando la respuesta por parte del candidato seleccionado. Si el alumno al que se le ha seleccionado para su realización acepta otra oferta que no es la de este profesor, aparecerá un mensaje de alerta en la columna de observaciones dentro de esta tabla de historial. En ese mensaje se notifica al profesor que debe elegir a otro candidato. Para ello, deberá hacer clic sobre la oferta, una vez en la página de la oferta, desotorgar al candidato actual y elegir a otro.

Página de listado de usuarios para usuario administrador

El administrador de la herramienta cuenta con una página en dentro de su menú principal con la opción de Usuarios para comprobar los usuarios actuales de la herramienta. En ella se dispone una tabla, con la misma estructura que en la página de inicio, donde se puede ver el nombre, apellidos, correo electrónico y el rol de los usuarios [Figura 4.30].



Usuario administrador

Filtrar por rama:
 --Elige un rol--

Listado actual de usuarios
 Número total: 7

Nombre A. v.	Apellido A. v.	Email A. v.	Rol A. v.
Luis	Olier	olier.luis@gmail.com	Alumno
Laura	Martinez Borluff	lauraborluff@gmail.com	Alumno
Adrián	González	adrianycia@hotmail.com	Alumno
Antonio	Morales	sakra3@hotmail.com	Alumno
Usuario	Prueba	usuariocodeprueba@gmail.com	Alumno
Laura	Martinez	lauraborluff@hotmail.com	Profesor
Administrador	Thais	thaisurjc@gmail.com	Administrador

Figura 4.30: Listado de usuarios

Esta tabla se puede ordenar por cualquiera de estos parámetros de manera alfabética ascendente o descendente. También se puede seleccionar que la tabla muestre sólo los usuarios de un determinado rol por medio de una barra desplegable. A su lado aparece una barra para buscar a un usuario en cuestión. Para ello el administrador puede introducir una palabra o parte de la palabra contenida en cualquiera de esos parámetros. El motivo de esta página es meramente informativo, para realizar los cambios que desee oportunos puede seleccionar la acción que necesite dentro del menú de Usuarios.

Página de cerrar sesión

Cuando un usuario desee terminar su sesión en nuestra aplicación, deberá dirigirse a la opción del menú principal situado horizontalmente y hacer clic con el ratón. Esta acción internamente lo hace es ejecutar un script en PHP. Al hacer clic desde cualquier página de nuestra sesión, nos redirigimos a este fichero en PHP. El contenido del fichero es el siguiente:

```
<?php
session_start();
session_unset();
session_destroy();
setcookie("id_extreme", "x", time()-3600, "/");
header("Location:../thais/formulario_login.php");
?>
```

El primer paso que se realiza es reanudar la sesión existente al principio del script mediante el comando `session_start()` de PHP. Después, con la llamada a la función `session_unset()`, lo que hacemos es liberar todas las variables de la sesión iniciada que están actualmente registradas. Como con esta última función, no se puede destruir completamente la sesión puesto que

desahabilitaría el registro de variables de sesión, como se indica en el manual de PHP [42]; por lo que, necesitamos convocar dos funciones más para finalizar la sesión correctamente. Con la función *session_destroy()* eliminamos toda la información registrada de la sesión, sin destruir ninguna de las variables globales asociadas con la sesión ni la cookie de sesión [42]. Para desconectar totalmente al usuario, la identidad de la sesión debe ser eliminado. Para ello, usando la función *setcookie()* para provocar su expiración. En esta función lo que hacemos es configurar el tiempo de la cookie para obligarla a finalizar. Una vez hechos todos estos pasos redirigimos a la página que contiene el formulario de inicio de sesión.

Offline

Una de las funcionalidades HTML5 que hemos agregado a la plataforma es la posibilidad por parte de los alumnos de ver la página de inicio y las páginas de ofertas en modo offline, es decir, sin tener conexión a la red. Habilitar páginas con acceso sin conexión es posible en navegadores modernos gracias al uso de la App Cache. En las siguientes líneas explicaremos la configuración seguida para ofrecer la funcionalidad de acceso sin conexión y los problemas encontrados.

■ Modificaciones en la configuración del servidor

Para dotar a nuestra plataforma con la funcionalidad de acceso offline hemos tenido que hacer algunos cambios en los ficheros de configuración del servidor Apache [46]. Más concretamente, hemos tenido que habilitar la lectura por parte de Apache del fichero *.htaccess* que por defecto viene desactivada [46]. Esto se realiza en:

```
/etc/apache2/sites-available/default
```

En la sección del fichero que muestre la configuración de la carpeta raíz de nuestro servidor [46], es decir, la carpeta donde guardamos todo lo que queremos mostrar via web (*/var/www/*), se debe modificar la línea que contiene *AllowOverride* y escribir:

```
AllowOverride All
```

Este cambio le indica a Apache que debe permitir a los ficheros *.htaccess* pasar por encima de las directivas previas [46]. Tras este paso, se deberá relanzar el servidor para que este cambio tenga efecto. A continuación creamos el fichero llamado *.htaccess* en la carpeta raíz con el texto:

```
AddType text/cache-manifest .manifest
```

■ Configuración del fichero `online.manifest`

Una vez hemos indicado al servidor cómo servir ficheros `.manifest`, creamos el fichero `offline.manifest` que será el que indicará qué recursos se deben poder servir sin conexión. Este fichero tiene un código con la siguiente estructura:

```
CACHE MANIFEST
```

```
CACHE:
```

```
#paths de los recursos que queremos que  
#se almacenen para poder visualizarse sin conexi\`on
```

```
#NETWORK:
```

```
#para recursos que no deseemos meter  
#en cache, como la p\`agina de login
```

```
#FALLBACK:
```

```
#indica las p\`aginas a las que se redirecciona si  
#no se encuentra el recurso cuando se esta sin conexi\`on
```

■ Carga del fichero `offline.manifest` en el navegador

Por último, para indicar al navegador la localización del fichero `offline.manifest`, debemos incluir el atributo `manifest` en la etiqueta `html` de la siguiente forma:

```
<!DOCTYPE html>  
<html manifest="../offline.manifest">
```

Siguiendo los pasos indicados anteriormente, cuando el usuario acceda a la página con el atributo `manifest`, el navegador descargará los recursos indicados en la parte de `CACHE` del fichero `offline.manifest` y comenzará a servir estos desde la caché local de ahí en adelante. En caso de que se pida un recurso que no esté en él, se carga `offline.html` como mensaje de advertencia.

■ Problemas encontrados

El hecho de que los recursos mencionados en la parte `CACHE` se sirven de forma local inmediatamente tras su descarga y hasta que se actualice el fichero `offline.manifest`, aunado con el hecho de que la página que nombra al fichero `offline.manifest` queda automáticamente guardada en la caché, hizo que la funcionalidad `offline` y la de cambio

de estilo CSS no funcionaran correctamente juntas. El navegador servía al usuario que cambiaba de estilo siempre el estilo que tenía guardado en la caché, sin tener en cuenta que la página a servir había cambiado. Para solucionar esto, tuvimos que forzar la carga del fichero `offline.manifest` desde una página alternativa (`inicio_offline.php`), que se cargase al mismo tiempo que se cargase `inicio.php` (esto último lo implementamos cargando `inicio_offline.php` por medio de un `iframe` dentro de `inicio.php`).

Otro inconveniente que tuvimos que hacer frente fue el hecho de que al crear una nueva oferta ésta debería estar disponible asimismo en modo sin conexión. Dado que un navegador que ya ha guardado App Cache no actualiza ésta hasta que vea un cambio en el fichero `offline.manifest`, debemos añadir una línea a este fichero cada vez que se crea una oferta en la plataforma. El detalle de este proceso se puede ver en el siguiente código:

```
# Añadimos la nueva oferta al offline.manifest
# para que esté disponible offline

# Sacamos el id de la oferta que se acaba de insertar
$oferta_identificador = "select MAX(num_ident) as num_ident
                        from ofertas where titulo=";
$oferta_identificador .= "'".$_POST['titulo']."'";
$conexion_bd = new mysqli($servidor,$usuarioBD,
                        $passwordBD,$baseDatos);
$resultado_query=$conexion_bd->query($oferta_identificador);
$num_identif_oferta=$resultado_query->fetch_assoc();

# Reescribimos el offline.manifest
$fileData = file('../offline.manifest');
$newArray = array();
foreach($fileData as $line) {
    $newArray[] = $line;
    if (substr($line, 0, 17) == '#Lista de ofertas') {
        $newArray[] = 'thais/oferta.php?num=
        '.$num_identif_oferta["num_ident"].'
        thais/oferta_offline.php?num=
        '.$num_identif_oferta["num_ident"];
    }
}
$result->free();
```

```
$mysqli->close();
file_put_contents("../offline.manifest", "");
$fh = fopen('../offline.manifest', 'a');
for( $i = 0; $i < sizeof($newArray); $i++ ){
    $newArray[$i] = trim($newArray[$i]);
    $content = $newArray[$i] . "\n";
    fwrite($fh, $content);
}
fclose($fh);
```

Por último, debemos mencionar que la funcionalidad offline sólo lo hemos implementado para el rol de alumno. Para los roles de profesor y administrado no lo hemos creído conveniente puesto que las acciones que pueden realizar profesores y administradores necesitan conexión con la base de datos (y por tanto acceso a la red). Dicho de otro modo, a un profesor sólo le interesará lo que tenga que ver con sus publicaciones (editarlas, ver sus notificaciones etc.). En cambio, sí es lógico que un alumno esté interesado en poder ver las ofertas publicadas en cualquier momento (online y offline); en modo sin conexión puede revisar las ofertas y cuando vuelva a tener conexión solicitar las que más le hayan gustado.

Página de estilo

En esta sección vamos a explicar cómo hemos diseñado el CSS de este proyecto. Todas las páginas están divididas en diferentes partes. La división que engloba a todas es el *background* o fondo que carga una foto. Esta división se desglosa a su vez en dos subdivisiones: el cuerpo y el *copyright*.

- **Copyright:** El copyright da estilo a la frase final de cada página, que contiene un enlace a la página de la Escuela.
- **Cuerpo:** El cuerpo define la región de la página que debe contener a sus subdivisiones: cabecera y *main*.

- *Cabecera:*

La cabecera, a su vez, contiene el título y el menú.

- *Título:* Para el título hemos recurrido a la implementación de CSS *@font-face*. Todos los *@font-face's* empleados los hemos sacado de la web *fonts2u.com*. Disponen de una gran variedad de estilos y permiten introducir una frase para ver cómo queda con cada fuente. Cuando eliges una fuente, ellos te ofrecen la

posibilidad de descarga del fichero para añadirla a las fuentes de tu ordenador o bien, utilizarla como *@font-face*. Cuando consigues el directorio con el *@font-face* que has elegido, solamente debes agregar los archivos con las extensiones: .eot, .svg, .ttf, .woff y el archivo de CSS. Tras esto se debe pegar el contenido de ese archivo de CSS o bien añadir ese archivo a la lista de CSS que debe consultar el navegador al cargar la página. Una vez hecho esto, le puedes dar el estilo que uno quiera, como el color, la posición dentro de la página, el tamaño, etc. Para el título contamos con tres *@font-face* distintos, uno para cada CSS. Para el CSS clásico hemos utilizado la fuente *SpringParty*, para el CSS con motivos de fantasía mostramos las letras de *WillyWonka*, y por último, para el estilo denominado escuela *SketchBlockBlod*.

- *Menú*: El menú principal es un menú horizontal, donde los usuarios disponen de los enlaces a las partes principales de la plataforma y dentro de este menú hay un submenú vertical. Poner este submenú no fue tarea fácil, puesto que para conseguir su posición correcta, tuvimos que hacer bastantes pruebas. Cuando un usuario se encuentra en una página dentro de las opciones de menú aparece remarcado con un cuadro de color, para recordar a la persona que está utilizando la herramienta dónde está. Además cuando se pasa el ratón por encima de cada una de las opciones, se marca también con el mismo formato, para una mejor visualización del puntero del ratón, que puede perderse entre todo el contenido de la página.

Dentro de la división de la cabecera, hemos colocado una imagen de bienvenida en la página de inicio (excepto en la de administrador). Esta imagen aparece cuando se carga dicha página y va desapareciendo poco a poco. Para conseguirlo hemos recurrido a las posibilidades que nos ofrece CSS3. Definimos el efecto: le hemos dado una duración de cinco segundos y le indicamos las veces que queremos que se ejecute. Luego para hacer que se desvanezca señalamos que tiene que pasar de una opacidad del 1 (siendo el 100 %) a 0. Además, queríamos que se quedase eliminada completamente, tras la ejecución del efecto, y mediante el siguiente comando lo conseguimos:

```
-webkit-animation-fill-mode: forwards;
```

Aparte de este efecto, también hemos redondeado la imagen para darle forma de círculo mediante una técnica similar a la utilizada en las tablas que detallaremos más adelante. En resumen, hemos ido redondeando las esquinas de la foto hasta tener esa forma. Intentamos utilizar una máscara con forma de círculo, pero tras mucho tiempo invertido en ello, nunca conseguimos que quedara igual de bien y, termina-

mos buscando una solución alternativa, como ésta de modificar las esquinas para resolver este problema. Aunque sea una solución menos elegante, resulta mucho más efectiva y simple.

- *Main:*

La otra subdivisión de cuerpo es *main*. Esta sección indica la posición dentro de la página, el color de fondo, blanco, y el ancho que ocupa, que sirve de margen para sus subdivisiones. Dependiendo de la información a mostrar tendrá un contenido u otro. Cuando se trata de una página con información (texto, tabla, etc.) creamos una división dentro de *main* para crear un cuadro centrado sobre el que mostrar toda la información. Para las secciones de elegir rama y buscar oferta, hemos utilizado un tabla sin bordes para que quedaran dispuestas una junto a la otra. Aunque tuvimos que cambiar los anchos de los botones, de `<select >` y de `<input >` para buscar para que tuvieran una separación central en el medio, que no se conseguía con la tabla. Nos dimos cuenta de esto al utilizar el navegador Mozilla Firefox, vimos que la barra de buscar era demasiado larga y se salía del área indicada para el contenido de la página. Esta subdivisión de contenido incluye el estilo que llevan todas las tablas de nuestra herramienta. El procedimiento que hemos hecho para dar este estilo a nuestras tablas la explicamos a continuación.

Lo primero fue determinar el ancho de la tabla a 100 %, porque si no el navegador pondría el que considerara por defecto y da un aspecto desordenado. Aplicamos unos bordes para que fuera más fácil la lectura de sus filas, con un píxel de grosor y quitamos el espacio entre bordes para que queden pegados los unos a los otros con la orden *boder-spacing*. La instrucción *boder-collapse* hace que los bordes se colapsen los unos sobre los otros manteniendo el efecto de borde de un píxel. Para que el texto no se pegara a los bordes se añadió la regla de *padding*, lo que hace que el texto se vea mucho más limpio dentro de la celda. En las tablas en las que contamos con un `<caption >` lo posicionamos siempre en la parte superior de la tabla. Redondeamos las esquinas superiores que contine la fila que define cada columna mediante las siguientes instrucciones de CSS3:

```
#tabla th:first-child{
    -moz-border-radius: 6px 0 0 0;
    -webkit-border-radius: 6px 0 0 0;
    border-radius: 6px 0 0 0;
}
#tabla th:last-child {
    -moz-border-radius: 0 6px 0 0;
    -webkit-border-radius: 0 6px 0 0;
```

```
border-radius: 0 6px 0 0;
}
```

Y además, sobre esta misma fila, cambiamos el color del fondo de la fila y el color de la fuente, para que quedara del todo resaltada y diferenciada respecto a las demás filas.

Durante esta fase de dar estilo a las tablas, nos encontramos con un error muy común. Cuando intentamos dar forma a nuestros bordes hay algunos navegadores web como Mozilla Firefox, Safari y Chrome que no muestran bien el ancho de los elementos que llevan la regla de *border-collapse: collapse*. Al establecer el borde a un píxel, como en nuestro caso, entre las dos primeras filas, las definidas con `<th >` y `<td >`, si a esta primera le ponemos un fondo distinto, aparece en un lado un píxel de diferencia entre ambas, quedando un borde más corto que el otro. Dependiendo del navegador este error aparecerá en un lado distinto. Según la información que encontramos en la red [43], este error es debido a una cuestión de redondeo que se reduce al problema de cómo representar medio píxel. Se podría afirmar que no se trata de un error *per se*, pero los navegadores no están de acuerdo, lo que provoca que en la práctica sí que se trate de un error [43]. Para solucionarlo, se puede aumentar el grosor del borde, con 2 píxeles ya no se notaría ese error. Pero si se empeña en que el borde sea de un píxel, tiene dos opciones. Una es aceptar este efecto, puesto que no se trata de un error grave ni del programador, forma parte de las diferencias entre los navegadores y la diferencia es muy pequeña. Y otra (la que hemos seguido) es manteniendo este borde de un píxel en el borde exterior y aumentando la separación entre celdas y aplicar un espacio entre ellas de cero. Esto hace que los bordes dentro de la tabla sean ligeramente más gruesos pero que el borde del recuadro exterior sea un píxel y se elimina esta diferencia de medio píxel entre título y celda. También otra alternativa a evitar esto, es no aplicando borde y/o color de fondo al título [43].

En las páginas que contienen un formulario la estructura cambia un poco. Se mantiene todo igual a excepción de la subdivisión contenido dentro de *main*. Estas páginas sólo constan de cabecera (con título y menú) y *main*. Esta última es la que incluye el formulario. El código de CSS para los formularios es el que sigue:

```
form.oferta{
    background: #ffffff;
    margin:auto;
    position:relative;
    width:490px;/*550px*/
    height:1050px;
    font-family: Arvo, serif;
    font-size: 14px;
    font-style: italic;
```

```
line-height: 24px;
font-weight: bold;
color: #367fb3; /*color azul enlace*/
text-decoration: none;
-webkit-border-radius: 10px;
-moz-border-radius: 10px;
border-radius: 10px;
padding:10px;
border: 1px solid #999;
border: inset 1px solid #333;
-webkit-box-shadow: 0px 0px 8px rgba(0, 0, 0, 0.3);
-moz-box-shadow: 0px 0px 8px rgba(0, 0, 0, 0.3);
box-shadow: 0px 0px 8px rgba(0, 0, 0, 0.3);

}
```

Lo más interesante de este código es lo relacionado a CSS3. Hemos redondeado las esquinas del borde del formulario de la misma forma que en la imagen de bienvenida. Este efecto lo hemos determinado tanto para los navegadores que funcionen con las librerías de *webkit* como con *moz*, los más utilizados por los usuarios de la web (Chrome, Safari y Firefox). Con *border-radius* conseguimos ese efecto de redondeado de los vértices. Si quisiéramos que estuvieran más pronunciadas, aumentaríamos el nivel de píxeles. También añadimos un ligero efecto de sombra en el formulario para destacarlo sobre el fondo con la regla *box-shadow*. Esta regla tiene varios parámetros. Los dos primeros controlan el *offset* de la sombra. Con un número positivo la sombra tomará una dirección hacia la derecha y arriba, en cambio, si es negativo, estará orientada hacia la izquierda y abajo. El siguiente dígito especifica el nivel de difuminado que queremos. Si el número es mayor, mayor difuminado y viceversa. El último parámetro determina el color de la sombra, el primer número dentro del paréntesis, indica el color rojo, el siguiente verde y el tercero el azul. El último número, que aparece con un valor decimal, indica el nivel de opacidad de la sombra. Si le damos un valor de 1 tendremos una sombra totalmente opaca, y 0 nada opaca. Estos valores representan una escala de tanto por ciento, es decir, el nivel 1 muestra la opacidad al 100 %, un 0.3 un 30 %, y así sucesivamente.

Obviamente, también hemos definido reglas para presentar el contenido del formulario de una manera limpia y ordenada.

En el la hoja de estilo llamada como *École* hemos añadido un *font-face* en el título de formulario (obtenido de la web *fonts2u.com*) y una imagen para el fondo. Nuestra intención era darle un aspecto de cuaderno de notas, para estar más dentro de la filosofía que engloba a una

escuela. Además las letras escogidas para el título son la típica fuente de letras con relieve y bordes dibujados y con un ligero guiño a un formato *Comic*, que parecen escritas por el mismo alumno. También resaltar que el botón para enviar el formulario en esta hoja de estilo es una foto, y como no podíamos hacer que cambiara de color cuando el ratón se situase encima, hemos hecho que cambiara el puntero del ratón de la flecha a una mano. Para así, hacer saber al usuario que en esa parte existe una acción y no es un adorno. En la hoja de estilo de 'Fantasía', hemos creado una franja de color en el área del título. En estas dos hojas, las esquinas las hemos dejado rectas para no sobrecargar la vista del formulario.

Para encajar bien el campo del curso, que definimos como un *input* de tipo *radio* en HTML, tuvimos que hacer una serie de pruebas. Al principio se mostraba por defecto en una columna. Nosotros preferíamos que todas sus opciones estuvieran dispuestas en una línea, así que modificamos el ancho de este elemento hasta que fue el deseado y con la orden de *display:inline*; lo conseguimos.

Para las hojas de estilo, primero creamos el formato clásico, y a partir de ésta, diseñamos las siguientes.

- **Estilo clásico:** En la implementación de esta hoja clásica nos inspiramos en las páginas de Apple. Si alguna vez han visitado su página, verán que utilizan tonos grises y esquinas redondeadas para las tablas, menú, etc.; para el texto utilizan tonos azules y grises más oscuros, y todo ello sobre un fondo blanco [Figura 4.31].

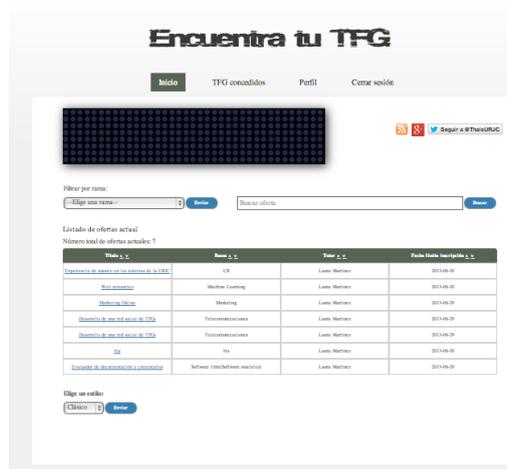


Figura 4.31: Hoja de estilo clásico

Siempre he considerado que esta página tenía un estilo muy elegante y simple, y quise intentar mostrar en este proyecto uno que siguiera esta línea. Desde esta hoja de estilo marqué la estructura de las otras dos, simplemente cambiando las imágenes de fondo, las fuente de las letras, las esquinas y los colores de los botones, enlaces, etc.

- Estilo Fantasía:** Para esta hoja queríamos dar un aspecto muy informal y con algunos elementos como el *font-face* utilizado para el título que llamaran la atención y marcaran una diferencia con respecto otros estilos similares ya vistos en otras páginas [Figura 4.32].



Figura 4.32: Hoja de estilo fantasía

- Estilo Escuela:** Para este estilo queríamos darle un toque estudiantil. Escogimos el color de fondo naranja, ya que es el color oficial de la ETSIT. Además elegimos unos *font-face* para los títulos de las páginas y de los formularios que siguieran con esa filosofía de escuela [Figura 4.33].

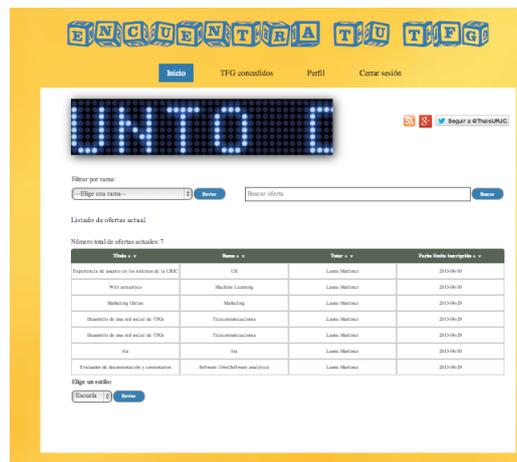


Figura 4.33: Hoja de estilo escuela

Para los formularios me apoyé bastante en lo explicado en [44], donde se explica cómo crear vistas elegantes en tus hojas de estilo con elementos con sombra, gradiente, etc. También me basé en las plantillas de ejemplo en [45]. Donde también te permiten utilizar herramientas en línea para crear mejor tus formularios, como constructores online, mensajes de error,

alternativas de disposición de los campos dentro del marco del formulario, herramientas para formalizar tu CSS, ejemplos de estilo a los distintos campos dentro de un formulario, etc. Toda la información que muestran es de uso libre, válida para todos los navegadores. Estas páginas aportan bastante información tocando cada una de las partes más importantes que puede constar un formulario, lo que resulta de gran ayuda, y evita tener que hacer más búsquedas de información sobre el estilo de cada uno de los elementos, lo que evita perder tiempo adicional.

4.3.4. Problemas encontrado durante el desarrollo de la herramienta

Durante el desarrollo de nuestra herramienta nos hemos encontrado con varias dificultades que entraremos a detallar a continuación.

Extensión *mysql* vs. *mysqli*

Al principio de la fase de desarrollo de la plataforma, cuando todavía no tenía la experiencia de ahora en PHP, tuve bastantes problemas a la hora de ejecutar mi código. El problema era los tutoriales que había seguido no eran del todo actuales. En ellos se usaban comandos para acceder a la base de datos por medio de *mysql* en vez de usar la interfaz *mysqli*. La extensión *mysql* está obsoleta y su uso está totalmente desaconsejado a partir de PHP 5.0. Será eliminada en un futuro según se indica en *php.net*. Al principio, no tenía constancia de esta advertencia y tuve bastantes problemas cuando ejecutaba mi código. Tras cargar dos veces una de mis páginas, ésta dejaba de funcionar, me devolvía una página completamente en blanco y me devolvía un error. La forma en la que se procedió para buscar una solución fue buscar ese error por Internet, pero toda la información que se encontraba no daba resultado. Después de varias semanas atascada con ese problema encontré la solución en un foro donde un usuario indicaba que ya estaba obsoleta. A partir de ahí el desarrollo fue mucho más rápido.

Mostrar imagen guardada en la base de datos

Este fue sin duda uno de los peores errores con los que nos hemos encontrado. Tras realizar varios simples ejemplos independientes de nuestra herramienta para poder probar el código de forma más sencilla. Siempre que intentábamos imprimir la foto nos salía la imagen típica del error de no encontrar la foto. El problema radicaba en el path y no en el código como pensamos durante la mayor parte del tiempo que estuvimos atascados.

Offline

Los problemas relacionados con el modo offline los hemos explicados en el apartado oportuno. Era más lógico comentarlos en ese punto donde mostrábamos toda la información necesaria. Simplemente recordar que los problemas relacionados con esta función vinieron como

resultado de dar la posibilidad a los usuarios de cambiar el estado de sus páginas. La página se quedaba guardada en caché y no mostraba los nuevos cambios. Para ellos hemos tenido que implementar el procedimiento por el cual se modifica la información dentro de `offline.manifest`, el fichero que se encarga de cargar las páginas sin conexión a la red.

Acentos en MySQL

Para permitir que el texto de los campos que introducimos en MySQL se guardara con acentos en la base de datos hemos tenido que cambiar el fichero `my.cnf`. El problema fue que, tras cambiar este fichero se nos cambió la configuración del servidor de MySQL, y no nos dejaba entrar. Para solucionarlo tuvimos que desinstalar el servidor de MySQL y volverlo a instalar.

Prueba en distintos navegadores

Durante el desarrollo de esta herramienta se ha ido probando en los navegadores Chrome, Firefox y Safari, disponibles en mi portátil. En todos ellos no he tenido ningún problema importante de compatibilidad. Lo único a resaltar es al cargar el CSS Fantasía. En Chrome, el fondo no se ajusta al tamaño de la ventana y deja un recuadro en blanco. En esto no pudimos resolverlo, puesto que es algo que tiene que ver dentro del intérprete del navegador y no con nuestro código. Lo único que pudimos hacer era darle un color de fondo por debajo de la foto en relación con los tonos del estilo para que se disimulara este efecto del navegador.

4.4. Resumen

Esta herramienta creada sirve para facilitar la tarea de concesión de los trabajos de fin de grado mediante un procedimiento transparente. Se creó para mejorar el nuevo protocolo que ha impuesto la Universidad en el último año para este tipo de concesiones. Con nuestra herramienta la Universidad puede tener constancia de las ofertas que publica cada profesor, qué grupos de trabajo entre profesores y alumnos se han formado y en qué trabajos. Así disminuir los casos en los que un alumno por no tener una nota media de expediente muy alta puede realizar, aún así, un trabajo que sea de su agrado.

Para esta herramienta se han creado tres tipos de roles: alumno, profesor y administrador. Las páginas que tienen los tres roles en común son: inicio, historial de T.F.G., página con la descripción detallada de la oferta, cambio de contraseña. Los roles de alumno y profesor tienen, además, en común una página que recoge las notificaciones que les pueden interesar.

El rol del alumno dispone de una página donde se muestra toda su información personal, CV, redes sociales, foto de perfil y una tabla que recoge sus solicitudes. También dispone una página para editar esa información y la funcionalidad de modo *offline*.

El rol profesor dispone de páginas para editar y crear ofertas y una página que recopila todas las ofertas que ha publicado.

El rol de administrador puede cambiar el rol de los usuarios, borrar usuarios, cambiar las contraseñas de los usuarios, editar la información de las ofertas, borrar ofertas y un página informativa con los usuarios que se han unido a la herramienta. Este rol sólo tiene la responsabilidad de controlar de la plataforma para que el funcionamiento de ésta sea el correcto.

En la siguiente tabla recogemos a modo de esquema las funcionalidades de HTML5 y CSS3 empleadas en este proyecto 4.1:

Lenguaje	Función
HTML5	Canvas, Offline, placeholder
CSS3	Animación de bienvenida, esquinas redondeadas, sombreado en elementos, títulos con <i>@fontface</i> , imagen de fondo adaptada a la página

Cuadro 4.1: Resumen funciones de HTML5 Y CSS3 utilizadas

Capítulo 5

Conclusiones

El gran auge que han tenido las redes sociales en los últimos tiempos ha hecho que se amplíe su enfoque a diferentes áreas y no sólo al ocio. Uno de esos enfoques que ha tenido mucho éxito ha sido el profesional con plataformas como LinkedIn o JobandTalent, donde se facilita la conexión con otras empresas y la labor de encontrar empleo. Además las redes sociales también se han ido extendiendo al terreno educativo facilitando la comunicación entre alumno y profesor, permitiendo un mejor seguimiento de la evolución del alumnado y continuando la enseñanza de las materias fuera de los límites de las aulas, el mejor ejemplo que podemos dar en el área educativa ha sido Moodle.

La herramienta Moodle en la ETSIT ha servido tanto para almacenar la información de los cursos como para publicar oferta de proyectos, prácticas y empleo. Eran los mismos profesores los encargados de anunciar dichos trabajos mediante una publicación en la página inicial para llegar al mayor número de alumnos posible. Este procedimiento no es del todo ideal deja a relucir la falta de una herramienta exclusivamente dedicada a ello. Un sitio donde consten todas las ofertas publicadas sin que éstas se mezclen con otras noticias sobre la universidad.

La universidad por su parte ha querido formalizar y oficializar las asignaciones de los TFG entre alumnos y profesores. Hasta ahora la forma de organizarse con estos trabajos era que el alumno y el profesor se ponían en contacto y decidían realizar el trabajo sobre un tema acordado entre ambos. Unas veces el profesor ofrecía directamente al alumno que considerara más interesado o capacitado para en el trabajo. O era el alumno el que acudía a un profesor que creía más experimentado en ese área para proponerle un proyecto. Como esto se hacía de forma extra-oficial, la Universidad muchas veces no tenía constancia de estos acuerdos y proyectos. Por este motivo desde este año ha creado un protocolo a seguir para la asignación de estos trabajos.

Nuestra herramienta ha sido diseñada para ayudar a los profesores y los alumnos de la ETSIT en el proceso de la asignación de TFG. Haciendo más llevadero el protocolo impuesto por la Universidad. Nuestra finalidad es facilitar un sitio donde alumnos y profesores se encuentren y puedan elegirse mutuamente para el desarrollo de un proyecto juntos. Los profesores pueden

publicar sus ideas de trabajos y disponen de información sobre los candidatos a estas ofertas para ayudar en su deliberación. A los alumnos, por su parte, se les da acceso a toda la información necesaria de las ofertas y la posibilidad de solicitar ese trabajo, haciendo que todos los alumnos tengan las mismas oportunidades. Formalizando de esta forma aquellos libres acuerdos que la universidad desea controlar.

5.1. ¿Qué cosas de las aprendidas durante la carrera he aplicado?

Para la realización de este proyecto he utilizado conocimientos aprendidos durante mi formación en estos años. En especial he empleado los conocimientos sobre creación de páginas webs.

Aprender a cómo configurar servidores como Apache ha influido positivamente durante el desarrollo del proyecto, y además, cómo ya lo había usado para otras prácticas durante la carrera, me dió una mayor seguridad durante todo el proceso de instalación.

Los conocimientos que adquirí sobre HTML básico me han ayudado bastante durante la implementación de la herramienta, así como también, las nociones aprendidas sobre MySQL. Cómo diseñar las tablas para guardar los datos, conectar con la base de datos y a realizar consultas a MySQL. También tener una experiencia previa a la hora de crear hojas de estilo para las páginas ha hecho que haya aprovechado más el tiempo en durante el desarrollo.

Cabe destacar que el haber estudiado varios lenguajes de programación me ha facilitado la tarea de empezar desde cero en el lenguaje de PHP, que era desconocido para mí.

5.2. ¿Qué cosas nuevas he aprendido?

Durante este proyecto he adquirido nuevos conocimientos. El principal de todos es el lenguaje de programación PHP. Algo bastante a destacar teniendo en cuenta que es uno de los lenguajes más utilizados en aplicaciones en Internet.

He ampliado mis conocimientos de HTML y he añadido en HTML5 que no había estudiado. Esta versión permite más funcionalidades como por ejemplo el canvas y el modo offline que se han realizado en este proyecto, que aportan mucho más dinamismo a las páginas. Además si a esto le añadimos la nueva experiencia adquirida en CSS3 que permite crear estilos más atractivos como las esquinas redondeadas, animaciones como la foto de bienvenida que va desapareciendo, añadir cualquier nueva fuente de letras, etc. Todo ello me ha proporcionado una gran experiencia a la hora de la creación de páginas web que hacen que tenga un nivel muy alto con el que competir en el mundo profesional.

También he ampliado de manera increíble mi manejo a la hora de hacer consultas de MySQL.

Ahora puedo realizar consultas mucho más complejas que las que aprendí durante los cursos de la carrera, una habilidad muy valorada por las empresas, ya que hoy en día toda la información está digitalizada y almacenada en bases de datos y hay saber cómo obtener los datos de éstas de una manera óptima y sin necesidad de malgastar los recursos.

Este proyecto me ha enseñado a cómo entender código ajeno y a la toma de decisiones en el desarrollo de la aplicación, buscando soluciones a los problemas que me iba encontrando, poniendo a relucir y fomentando mi capacidad de ingeniero.

5.3. Problemas en la planificación

Este proyecto de fin de carrera se pensó, en un principio, defenderlo en el mes de marzo, pero debido a unos contratiempos en la planificación se retrasó hasta el mes de junio.

Uno de las dificultades que nos encontramos fue respecto dónde implantar la herramienta. En un principio se optó por la plataforma Moodle, muy usada en la Escuela. Se invirtió bastante tiempo en el conocimiento de esta herramienta, se hizo un estudio a fondo de todos los posibles plugins que dispone la organización y ver cómo se podría juntar aquéllos que más nos interesara. Descubrimos que de este modo aún quedaba incompleta la herramienta. También optamos por la creación de un plugin propio para Moodle, pero lo descartamos con el tiempo puesto que de esta forma dependeríamos de la decisión de la Universidad para implantarlo o no en la herramienta oficial y como queríamos que desde un principio esta aplicación tuviera un uso real optamos por otras opciones.

Encontramos otra herramienta en la línea de Moodle, pero con muchas más funcionalidades sociales, la plataforma Mahara. Nos centramos en ella, porque veíamos que sería muy positivo para los alumnos disponer de estas características sociales dentro de una plataforma que conectada con Moodle podría facilitar las tareas de las materias por medio de su uso. Dedicamos bastante tiempo al conocimiento de esta plataforma tanto por dentro como por fuera e, incluso, nos pusimos en contacto con la organización del sitio para resolver dudas para hacer posible nuestra finalidad. Pero volvimos a darnos cuenta que también, para esta opción, necesitaríamos el consentimiento de la Universidad y es bastante estricto en la agregación de nuevos módulos al Moodle, por motivos de seguridad. Lo que nos llevó, finalmente, a la idea de crear nuestra aplicación de forma completamente independiente. Los errores anteriores nos llevaron demasiado tiempo lo que provocó la ralentización de la planificación en el desarrollo de este proyecto.

5.4. Líneas futuras

En este apartado propondremos ideas sobre el posible desarrollo que puede tener la aplicación web que presentamos en un futuro. Estas ideas pueden tomarse por otros profesores y

alumnos que lo deseen como futuros proyectos de fin de carrera o trabajos fin de grado. Destacamos las siguientes líneas:

- **Integración con Moodle:** Después de que se haya demostrado su utilidad, si la herramienta tiene el éxito esperado, se podría agregar al Moodle de la ETSIT. Como la universidad tiene una postura muy estricta en cuanto a nuevos módulos a agregar a esta plataforma, sólo sería lógico realizar esta tarea una vez probado su éxito. Para la integración se debería seguir el esquema de plugins para Moodle y añadir un enlace en la página principal para que los alumnos puedan acceder a la herramienta desde ahí. No resultaría muy complicado, puesto que se ha tenido en cuenta en el desarrollo de la herramienta.
- **Creación de un perfil de empresa:** Otra de las futuras implementaciones a la herramienta sería permitir a empresas publicar ofertas para los estudiantes de ingeniería de telecomunicación. Para ello deberían disponer de un perfil donde señalar los rasgos y filosofía de la compañía para información de los alumnos, por ejemplo, siguiendo la línea del que disponen en LinkedIn.
- **Integración con el COIE:** Dentro de este carácter empresarial, podría añadirse con el COIE de la universidad o servir de sustituto para los alumnos de la ETSIT. De este modo se quitaría volumen de trabajo a este centro y se daría una atención más focalizada a estos alumnos, ya que el COIE difunde ofertas para todos los alumnos de todas las escuelas y facultades que forman la universidad.
- **Funcionalidad para realizar entrevistas:** se podría implementar una funcionalidad para que en el caso de que una empresa quiera realizar una entrevista en línea de los candidatos de sus ofertas si lo ve necesario. Se podrían poner en contacto para determinar una fecha y realizar la entrevista desde nuestro sitio por medio de una videoconferencia.
- **Adjuntar contenido multimedia del trabajo una vez finalizado:** Otra opción que indicamos es la de agregar contenido multimedia a los trabajos una vez finalizados y presentados. El tutor o el alumno que realizó el proyecto podría tener la posibilidad de subir vídeos donde se explique el trabajo y la presentación. Así los alumnos que se interesen por ciertos trabajos puedan conocer todos los detalles por medio de la herramienta. Además podrían añadirse álbumes de fotos dedicados a las distintas fases del desarrollo, las diapositivas de la presentación, enlaces para visitar el trabajo (en caso de ser aplicaciones web), enlace directo para la descarga del trabajo, etc.

Apéndice A

Manual de usuario para THAIS

En esta sección redactaremos un manual de uso para cada uno de los roles de usuario en nuestra red. En cada uno de los roles explicaremos en detalle todas las funciones que pueden realizar en nuestra herramienta, paso por paso explicados desde el registro hasta que cierre la sesión.

A.1. Usuario alumno

A continuación detallaremos el tutorial de uso de nuestra herramienta para los alumnos. Explicaremos el funcionamiento página por página.

A.1.1. Registro y creación de cuenta de usuario en THAIS

Para aquel alumno que se inicie en el uso de nuestra aplicación, lo primero que deberá hacer será dirigirse a la página http://pfc-lborlaff.libresoft.es/thais/formulario_login.php [ver figura A.1]. Puede registrarse y crear una cuenta de usuario haciendo clic sobre el enlace de 'Registrarse' que aparece debajo del formulario para el inicio de sesión.



Figura A.1: Página de inicio de sesión de THAIS

Pre-registro

Desde ese enlace, se redirigirá automáticamente a la página contenida en la URL http://pfc-lborlaff.libresoft.es/thais/formulario_pre_registro.php [ver figura A.2]. En ella aparecerá un formulario con un campo. En ese campo, se pide al usuario que introduzca su correo electrónico, donde recibirá un email para la creación de la cuenta. Es posible que este correo se archive en la carpeta de correos no deseados o *spam*, por lo que pasados unos minutos, si el usuario no recibe el correo, compruebe estas bandejas de entrada.

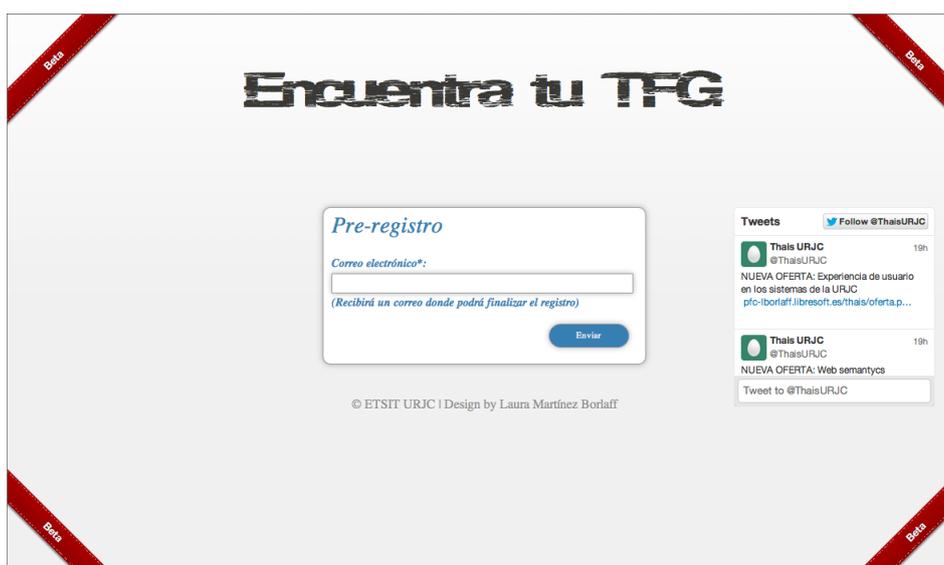


Figura A.2: Página de pre-registro de THAIS

En este correo se enviará un enlace para la activación de la cuenta [ver figura A.3]. El

alumno deberá dirigirse a la página de ese enlace.



Figura A.3: Ejemplo de mail para finalizar el proceso de registro en THAIS

Registro

Una vez dentro, en este nuevo formulario procedemos al registro como usuarios. En él nos piden los datos: nombre, apellidos, el email (que ya viene con el dato introducido anteriormente de manera automática) y la contraseña que queremos usar para nuestra cuenta dentro de esta herramienta, que se deberá introducir dos veces [ver figura A.4].



Figura A.4: Página de registro de THAIS

Se recomienda por motivos de seguridad no utilizar la misma contraseña que en el correo. Nuestra aplicación consta con los mecanismos de seguridad propios de cualquier web, pero como usuarios, debemos de ser prudentes a la hora de poner contraseñas en las cuentas donde podamos dar nuestro correo como cuenta de usuario; nunca poner la misma clave que se usa en el correo, porque nunca sabemos a ciencia cierta quién utiliza esos datos luego.

Una vez rellenados todos los campos, pulsamos el botón de enviar. Si alguno de los campos introducidos no es correcto nos devolverá un mensaje de error diciendo los que repasemos. Si todos son correctos, nos redireccionará a la página para iniciar sesión.

A.1.2. Inicio de sesión

Volviendo a la página del principio del primer paso [ver figura A.1], ubicada en http://pfc-lborlaff.libresoft.es/thais/formulario_login.php. Como se puede observar, a la izquierda del formulario aparece el torrente de tweets. En él los usuarios pueden leer el historial de todas las entradas a esta red social. A esta red se envía un nuevo tweet con cada nueva oferta que se publique.

En esta página, para iniciar nuestra sesión en THAIS, introducimos la dirección de correo que se uso en el registro, la contraseña y pulsamos el botón de enviar.

A.1.3. Página de inicio

Una vez confirmada que los datos son correctos, nos direccionara a la página de inicio de THAIS. En ella vemos que disponemos de un menú con las opciones: inicio, TFG concedidos, perfil y cerrar sesión. La opción de perfil engloba otras opciones: ver, editar, notificaciones y cambiar contraseña. Podemos ver este submenú situando la flecha del ratón sobre la opción perfil. De ella, aparecerá un menú desplegable con esas opciones.

En el letrero luminoso se podrá leer notificaciones sobre las ofertas que están a una semana de finalizar su plazo de inscripción [ver figura A.5]. En caso de haber ofertas a punto de expirar se imprimirán los títulos de dichas ofertas. Si no hay ofertas que expiren próximamente se mostrará el mensaje informativo sobre ello. Al lado de este letrero luminoso, a la derecha, podemos ver los botones para ver el RSS y añadir la cuenta de la herramienta a nuestras redes de GooglePlus y Twitter. En el fichero RSS contiene el título, la descripción y la fecha de publicación de todas las ofertas de la plataforma. Además de la fecha de creación del fichero RSS.



Figura A.5: Página de inicio de THAIS (Parte I)

En la línea siguiente vemos dos barras [ver figura A.5]. A la izquierda un desplegable donde se puede elegir una rama por la que filtrar las ofertas a mostrar. Y a la derecha una barra donde buscar una oferta por coincidencias.

Más abajo [ver figura A.6], encontramos la tabla de ofertas. En la línea anterior a la tabla se nos muestra el número total de ofertas.

Filtrar por rama:

--Elige una rama--

Listado de ofertas actual
Número total de ofertas actuales: 7

Título $\wedge \vee$	Rama $\wedge \vee$	Tutor $\wedge \vee$	Fecha límite inscripción $\wedge \vee$
Experiencia de usuario en los sistemas de la URJC	UX	Laura Martínez	2013-06-30
Web semantics	Machine Learning	Laura Martínez	2013-06-30
Marketing Online	Marketing	Laura Martínez	2013-06-29
Desarrollo de una red social de TFGs	Telecomunicaciones	Laura Martínez	2013-06-29
Desarrollo de una red social de TFGs	Telecomunicaciones	Laura Martínez	2013-06-29
bla	bla	Laura Martínez	2013-06-30
Evaluador de documentación y comentarios	Software libre(Software analytics)	Laura Martínez	2013-06-29

Elige un estilo:

Clásico

Figura A.6: Página de inicio de THAIS (Parte II)

La tabla nos muestra las ofertas junto con información de título, rama, tutor y fecha límite de inscripción. Al lado de cada uno estas columnas están los símbolos \wedge y \vee , con ellos podremos ordenar la tabla por ese campo de manera alfabética ascendente o descendente, o también, por fecha descendente o ascendente. Todos los títulos de las ofertas son enlaces a páginas donde se detalla toda la información sobre ellas.

Debajo de la tabla de ofertas, tenemos otro menú desplegable donde podemos elegir el estilo que más nos guste. Se puede elegir entre las opciones: Clásica, Fantasía o Escuela.

A.1.4. Página de una oferta

Pulsando sobre el enlace de una de las ofertas llegamos a una página como por ejemplo la que puede verse en A.7.



Figura A.7: Página de una oferta en THAIS

Como se puede ver, contiene toda la información sobre la oferta. Arriba de el campo de título aparece un enlace con la palabra 'Aplicar'. Si un alumno desea optar a este trabajo deberá hacer clic en este enlace. Si ya ha enviado su solicitud, cuando vuelva a entrar en la página de la oferta se verá en vez del enlace, un texto corto notificando que ya se había inscrito. Si la oferta ha expirado, pondrá el texto informando sobre ello y tampoco se mostrará el link para solicitar.

Al final de la página, aparecen los botones para hacer +1 en GooglePlus o publicar la oferta en el Twitter del alumno.

A.1.5. Página de TFG concedidos

Si vamos a la opción del menú TFG concedidos veremos una página con un contenido similar al de la figura A.8]

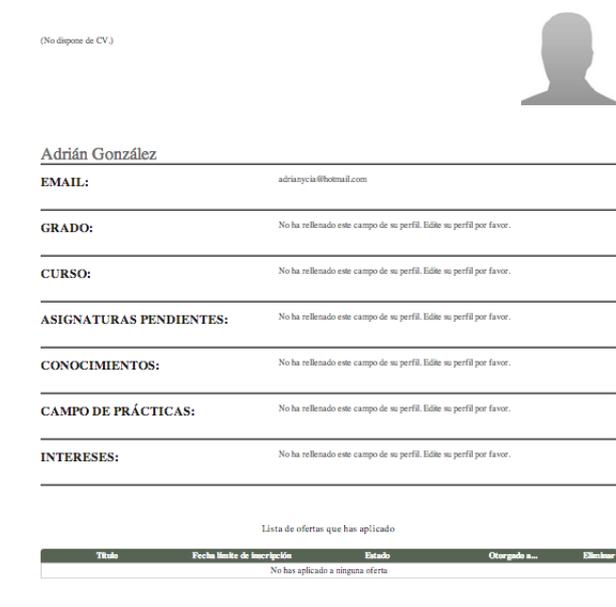
Título	Rama	Tutor	Departamento	Alumno elegido
hello	hello	Laura Martínez	hel	Laura Martínez Borlaff

Figura A.8: Página de TFG concedidos en THAIS

Vemos una tabla con los campos título, rama, tutor, departamento y alumno elegido. Las ofertas que aparecen en esta tabla son ofertas que ya han sido asignadas a otros alumnos. El carácter de esta página es meramente informativo. Las ofertas de esta tabla también están enlazadas a la página de su descripción.

A.1.6. Perfil de usuario

Para ver el perfil del usuario debemos ir a Perfil→Ver. En él vemos la página de la figura A.9.



(No dispone de CV.)

Adrián González

EMAIL: adrianycia@hotmail.com

GRADO: No ha rellenado este campo de su perfil. Edite su perfil por favor.

CURSO: No ha rellenado este campo de su perfil. Edite su perfil por favor.

ASIGNATURAS PENDIENTES: No ha rellenado este campo de su perfil. Edite su perfil por favor.

CONOCIMIENTOS: No ha rellenado este campo de su perfil. Edite su perfil por favor.

CAMPO DE PRÁCTICAS: No ha rellenado este campo de su perfil. Edite su perfil por favor.

INTERESES: No ha rellenado este campo de su perfil. Edite su perfil por favor.

Lista de ofertas que has aplicado

Título	Fecha límite de inscripción	Estado	Otorgado a...	Eliminar
No has aplicado a ninguna oferta				

Figura A.9: Perfil de usuario alumno en THAIS

De arriba hacia abajo y de izquierda a derecha, vemos un pequeño texto indicando que no se dispone de CV para descargar. Cuando el alumno adjunte ese archivo en su perfil aparecerá un enlace en el que poder descargar ese documento. A lado aparece la foto que también puede ser subida por el usuario. Después, los campos que consta el perfil. Más abajo una tabla donde se mostrarán las ofertas que el usuario vaya solicitando, con sus respectivos enlaces a sus páginas de información. Si al usuario se le ha otorgado una oferta, sólo aparecerá ese registro en la tabla.

A.1.7. Editar perfil de usuario

Para editar el perfil de usuario nos dirigimos a Perfil →Editar, que nos llevará a la página mostrada en la figura A.10.

Formulario editar perfil

Nombre*:
 Dirección:
 Apellidos*:
 Correo electrónico*:
 Grado:
 Curso:
 Asignaturas pendientes:
 Conocimientos:
 Campo de prácticas:
 Twitter:
 Google+:
 LinkedIn:
 Foto de perfil:
 Curriculum vitae:

Actualizar

Figura A.10: Editar perfil de usuario alumno en THAIS

En el formulario que aparece se puede editar los campos de nombre, apellidos, correo electrónico, grado, curso, asignaturas pendientes, conocimientos, campo de prácticas, intereses del alumno, cuenta de Twitter, Google+, LinkedIn, foto de perfil y documento CV (en .pdf).

Destacar que para las redes sociales se deberá rellenar con el nombre de usuario en Twitter, la URL al perfil de GooglePlus y la URL de su perfil público en LinkedIn.

A.1.8. Notificaciones de usuario

Para ver el historial de las notificaciones para el usuario dentro de THAIS deberá dirigirse a Perfil→Notificaciones. En ella se verá información parecida a la siguiente: A.11.

29-05-2013	¡Nueva oferta!	Evaluador de documentación y comentarios
29-05-2013	¡Nueva oferta!	bla
29-05-2013	¡Nueva oferta!	jastidelf
29-05-2013	¡Nueva oferta!	hofo
29-05-2013	¡Nueva oferta!	Desarrollo de una red social de TTG
29-05-2013	¡Nueva oferta!	Desarrollo de una red social de TTG
29-05-2013	¡Nueva oferta!	Marketing Online
30-05-2013	¡Nueva oferta!	Experiencia de usuario en los sistemas de la URJC
30-05-2013	¡Nueva oferta!	Web semantycs

Figura A.11: Notificaciones de usuario alumno en THAIS

Se recibirán avisos sobre nuevas ofertas o estados sobre las ofertas solicitadas.

A.1.9. Cambiar contraseña

Accediendo a la página contenida en Perfil →Cambiar contraseña. Veremos un formulario donde introducir la nueva contraseña que queremos en nuestra cuenta en THAIS. Debemos introducirla dos veces A.12.

El formulario tiene un título "Cambiar password" en azul. Debajo hay un campo de texto con el label "Contraseña*" y el placeholder "Escribe la nueva contraseña". A continuación, otro campo de texto con el label "Repite la contraseña*" y el placeholder "Repite la nueva contraseña". En la parte inferior derecha del formulario hay un botón azul con el texto "Cambiar".

Figura A.12: Cambio de contraseña de usuario alumno en THAIS

A.1.10. Cerrar sesión

Tan sólo pulsando la opción de Cerrar sesión contenida en el menú principal, se cerrará correctamente nuestra cuenta en THAIS y volveremos a la página con el formulario de inicio de sesión.

A.2. Usuario profesor

Para establecer un usuario profesor, el profesor en deberá proceder de la misma forma que un usuario alumno para el registro. Una vez finalizado su registro, deberá informar al usuario administrador para que le otorgue poderes de profesor y recibirá un email informando sobre su nuevo rol.

Para iniciar sesión también procederá de la misma manera que lo explicado en usuario alumno.

A.2.1. Página de inicio

La página de inicio es muy parecida a la de alumno, a excepción del letrero luminoso [figura A.13]. A parte de eso, el profesor presenta las mismas funciones de filtrado por rama, búsqueda de oferta, la tabla de las ofertas, la selección de estilo, archivo RSS y añadir a las redes sociales el perfil de THAIS. Para información más detallada sobre sus funciones dirigirse al apartado página de inicio de usuario alumno.

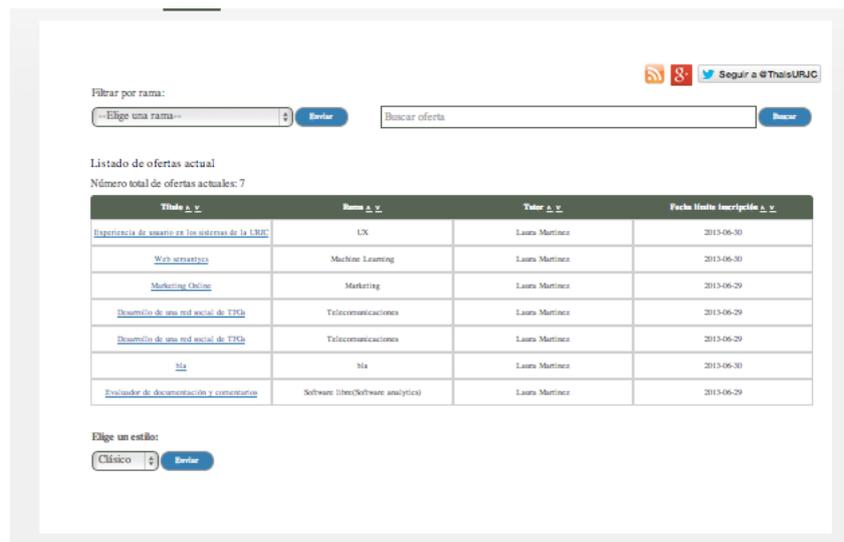


Figura A.13: Página de inicio de usuario profesor en THAIS

A.2.2. Página de TFG concedidos

Esta página es exactamente igual que en la sesión de alumno. Para más detalles sobre sus funciones dirigirse al apartado página de TFG concedidos de usuario alumno.

A.2.3. Página para crear una nueva oferta

Para agregar una nueva oferta a la base de datos, se deberá dirigir, el usuario profesor, a la opción del menú principal llamada Nueva oferta [figura A.14].

Formulario nueva oferta

Título*:
Título de la oferta

Rama*:
Rama de la especialización de la oferta

Tutor*:
Laura Martínez

Email de contacto*:
laurambrill@hotmail.com

Página personal:
URL de la página personal del tutor

Departamento*:
Departamento al que pertenece la oferta

Descripción*:
Descripción de la oferta

Conocimientos:
Conocimientos requeridos

Tecnologías:
Tecnologías que se usarán

Deadline: 31 de 6 de 2013

Enviar

Figura A.14: Página para crear una nueva oferta de usuario profesor en THAIS

Se deberá rellenar el formulario con los campos que aparecen para crear la oferta. No todos los campos son obligatorios. Como mínimo para crear una oferta se necesitan los datos: título, rama, tutor, email de contacto, departamento descripción y fecha límite de inscripción.

A.2.4. Página mis ofertas

Los profesores cuentan con una página donde tener un seguimiento más centrado de sus ofertas publicadas. Esta página se encuentra dentro del menú perfil, en Perfil→Mis ofertas [figura A.15].

Ofertas por asignar
Número total: 6

Título	Fecha límite inscripción	Número de candidatos
Web semestral	2013-06-30	1
Marketing Online	2013-06-29	0
Desarrollo de una red social de TFGs	2013-06-29	1
Desarrollo de una red social de TFGs	2013-06-29	0
No	2013-06-30	0
Evaluador de documentación y comentarios	2013-06-29	1

Historial de TFGs que has concedido

Título	Alumno elegido	Observaciones
Asíadef	Luisito Puerto	Esperando contratación por parte del alumno
No	Laura Martínez Barja	-

Figura A.15: Página de mis ofertas de usuario profesor en THAIS

Esta página está dividida en dos partes: una tabla con sus ofertas que todavía falta por encontrar alumno y otra tabla con las ofertas ya concedidas. En la página de cada una de estas ofertas se podrán realizar las acciones de selección de candidatos. Cuando un profesor seleccione a un usuario para una oferta, esa oferta pasará de la primera tabla a la segunda y dependiendo del estado en el que se encuentre el procedimiento de la concesión de la oferta, se imprimirá un mensaje en la columna de observaciones. Si se ha seleccionado un alumno y está esperando su respuesta, o si el alumno ha aceptado otra oferta, mostrará el mensaje oportuno.

A.2.5. Página de una oferta

Esta página difiere de la mostrada para el usuario alumno. En el rol del profesor se dan las acciones de editar la oferta, eliminar la oferta o en la tabla que aparece abajo selección del candidato [figura A.16].

Título: Web semantycs
Rama: Machine Learning
Tutor: Laura Martínez
Email de contacto: lauramborlaff@hotmail.com
Departamento: AI
Descripción: Ejemplo de oferta
Conocimientos:
Tecnologías:
Fecha límite de inscripción: 2013-06-30
(Última actualización: 2013-05-30 00:18:53)

Candidatos

Nombre	Email	Elegir
Laura Martínez Borlaff	lauraborlaff@gmail.com	Seleccionar

[R+1](#) | [0](#) | [Twitter](#) | [0](#)

© ETSIT URIC | Design by Laura Martínez Borlaff

Figura A.16: Página de oferta de usuario profesor en THAIS

En la acción de editar oferta nos redirigiremos a un formulario como el de nueva oferta con los campos ya introducidos escritos en él. Debemos rellenar aquellos campos con los datos que queramos añadir o modificar.

En la tabla de la parte inferior de la página aparece una lista con los candidatos que han solicitado realizar el trabajo. Éstos están enlazados a sus perfiles, donde el profesor podrá obtener la información sobre la preparación del postulante. Y seleccionar el que le parezca adecuado desde esta misma tabla.

A.2.6. Página de notificaciones

En esta página que podemos encontrar en Perfil→Notificaciones, se nos muestran las noticias referentes a las ofertas publicadas por el usuario profesor. Nos informan si un alumno se ha apuntado como candidato, si un alumno ha aceptado otra oferta en vez de la nuestra o si el alumnos elegido ha aceptado realiza nuestra oferta. [Figura A.17]



Figura A.17: Página de notificaciones de usuario profesor en THAIS

A.2.7. Cambiar contraseña

Esta página es análoga a la de alumno. Por favor, dirigirse al apartado sobre cambiar contraseña de usuario alumno.

A.3. Usuario administrador

Para este usuario incluiremos una sección para detallar la preparación del entorno en el servidor donde se ejecutará además de las secciones explicando todas las funciones que puede realizar.

A.3.1. Preparación del entorno

Para el correcto funcionamiento de la herramienta se necesita una serie de requisitos en cuanto a software para preparar el entorno donde se ejecutará.

Primero instalaremos el servidor. Nosotros escogimos Apache2 puesto que la combinación de Apache2, MySQL y PHP sobre Ubuntu funciona a la perfección. Para instalar Apache, debemos abrir un terminal y escribir: `sudo apt-get apache2`. Subimos el servicio Apache mediante: `sudo service apache2 start`. En un navegador ingresamos la dirección: `http://localhost`. Si aparece el mensaje "It works" en pantalla, estará instalado correctamente. [28]

Nuestra herramienta almacena toda la información en tablas de MySQL. Se debe contar con esta aplicación para que funcione como es deseado. Ingresamos la siguiente instrucción en el terminal: `sudo apt-get install mysql-server` [28]. Con esto se instalará el servidor de MySQL. En el proceso, se nos pedirá definir la contraseña de root [28]. Esta contraseña deberá indicarse en el fichero `config.php`. Ahora el cliente, para ello introducir: `sudo apt-get install mysql-client` [28]. Y con esto queda instalado MySQL aunque todavía no ha acabado con esta aplicación. Para que se pueda introducir texto con acentos y se guarde correctamente en las tablas debemos seguir los siguientes pasos. Primero localizar el fichero `my.cnf` de MySQL por medio de la instrucción `locate my.cnf`. Una vez encontrado, debemos editarlo del siguiente modo:

- En la parte de `[client]` añadir la línea: `default-character-set=latin1`
- En el apartado de `[mysqld]` agregar:

```
default-character-set=latin1
character-set-server=latin1
collation-server=latin1_spanish_ci
init_connect='SET collation_connection=latin1_spanish_ci'
```

```
init_connect='SET NAMES latin1'
```

- Finalmente en el apartado [mysql] añadir la siguiente línea: `default-character-set=latin1`

Con esto ya tenemos instalado y configurado MySQL, listo para trabajar.

Como nuestra herramienta está escrita en PHP necesitamos tener instalado este lenguaje. Se consigue por medio de la instrucción: `sudo apt-get install php5` [28]. Debemos de instalar, también `sudo apt-get install php5-curl`, para que funcione el envío de tweets automáticos en la cuenta de la herramienta cuando se publique una nueva oferta.

Para finalizar, sólo queda instalar el gestor de archivos 'pear', necesario para ejecutar las librerías de conexión SMTP. Para ello, introducimos en el terminal `sudo apt-get install php-pear`. Comprobamos que no hay que añadirlo al path con escribiendo en el terminal `pear version` e instalamos la librería Mail mediante: `sudo pear install Mail`. Nótese que la letra 'M' de Mail es mayúscula. Y a continuación instalamos la librería SMTP: `sudo pear install Net_SMTP`. Después de estos pasos deberemos relanzar el servidor Apache para que vuelva a cargar PHP: `sudo apachectl restart` [47].

Tras este último paso guardamos en la carpeta raíz de nuestro servidor, en el caso de Ubuntu, `/var/www` el directorio de nuestra herramienta. Y ya tenemos todo listo para empezar a disfrutar de nuestra herramienta.

A.3.2. Registro y creación de cuenta

El usuario administrador no necesita registro para la creación de la cuenta, ya dispone de una por defecto dentro de la herramienta, así como de una dirección de correo propia, que se creó durante su desarrollo.

Simplemente, tras su instalación, deberá dirigirse a la página de inicio de sesión e introducir la cuenta de correo y la contraseña preestablecida.

A.3.3. Página de inicio

La página de inicio para el usuario administrador es bastante parecida a la del profesor. [Figura A.18]

Usuario administrador

Seguir a @ThaisURJC

Filtrar por rama:
 --Elige una rama-- [Enviar] Buscar oferta [Buscar]

Listado de ofertas actual
 Número total de ofertas actuales: 7

Título ▲ v	Rama ▲ v	Tutor ▲ v	Fecha límite inscripción ▲ v	Modificar	Eliminar
Experiencia de usuario en los sistemas de la URJC	UX	Laura Martínez	2013-06-30	[Editar]	[Borrar]
Web semánticas	Machine Learning	Laura Martínez	2013-06-30	[Editar]	[Borrar]
Marketing Online	Marketing	Laura Martínez	2013-06-29	[Editar]	[Borrar]
Desarrollo de una red social de TFGs	Telecomunicaciones	Laura Martínez	2013-06-29	[Editar]	[Borrar]
Desarrollo de una red social de TFGs	Telecomunicaciones	Laura Martínez	2013-06-29	[Editar]	[Borrar]
bla	bla	Laura Martínez	2013-06-30	[Editar]	[Borrar]
Evaluador de documentación y comentarios	Software libre (Software analytics)	Laura Martínez	2013-06-29	[Editar]	[Borrar]

Elige un estilo:
 Clásico [Enviar]

Figura A.18: Página de inicio de usuario administrador en THAIS

La única diferencia es que en la tabla de las ofertas aparecen directamente en la misma fila de cada oferta las acciones para editar su información o borrarla de la base de datos. Para más información ver el punto homólogo detallado en usuario alumno.

A.3.4. Página TFG concedidos

Esta página es equivalente a la explicada en el usuario alumno, pero añadiendo la opción de borrar una oferta de la base de datos desde la tabla, sin necesidad de tener que abrir la página de la oferta. Esta acción puede interesar cuando se desee hacer una limpieza de las ofertas ya otorgadas muy antiguas o en algún posible caso de error. [Figura A.19]

Usuario administrador

Historial de TFGs concedidos

Título	Rama	Tutor	Departamento	Alumno elegido	Eliminar oferta
hello	hello	Laura Martínez	hel	Laura Martínez Borlaff	[Borrar]

Figura A.19: Página de TFG concedidos de usuario administrador en THAIS

A.3.5. Página de una oferta

El usuario administrador disfruta de los mismos privilegios sobre las ofertas que el tutor de la oferta. Además puede borrar la candidatura de un usuario. Sólo puede proceder en caso de error en la herramienta y no interponerse en el procedimiento de selección en cualquier trabajo ofertado. [Figura A.20]

Usuario administrador

Título: Web semantycs

Rama: Machine Learning

Tutor: Laura Martinez

Email de contacto: lauramborlaff@hotmail.com

Departamento: AI

Descripción: Ejemplo de oferta

Conocimientos:

Tecnologías:

Fecha límite de inscripción: 2013-06-30

(Última actualización: 2013-05-30 00:18:53)

Candidatos

Nombre	Email	Elegir
Laura Martinez Borlaff	lauraborlaff@gmail.com	Seleccionar Borrar

Figura A.20: Página de una oferta de usuario administrador en THAIS

A.3.6. Página para crear una nueva oferta

Esta página es análoga a la de usuario profesor, para más información leer este punto en el profesor.

A.3.7. Página con listado de usuarios

Esta página se encuentra en Usuarios→Listado. En ella tenemos un desplegable por medio del cual filtrar el rol de los usuarios a mostrar en la tabla, y una barra para la búsqueda de un usuario en concreto por nombre, apellidos o email. Esta página tiene una finalidad informativa exclusivamente [figura A.21].

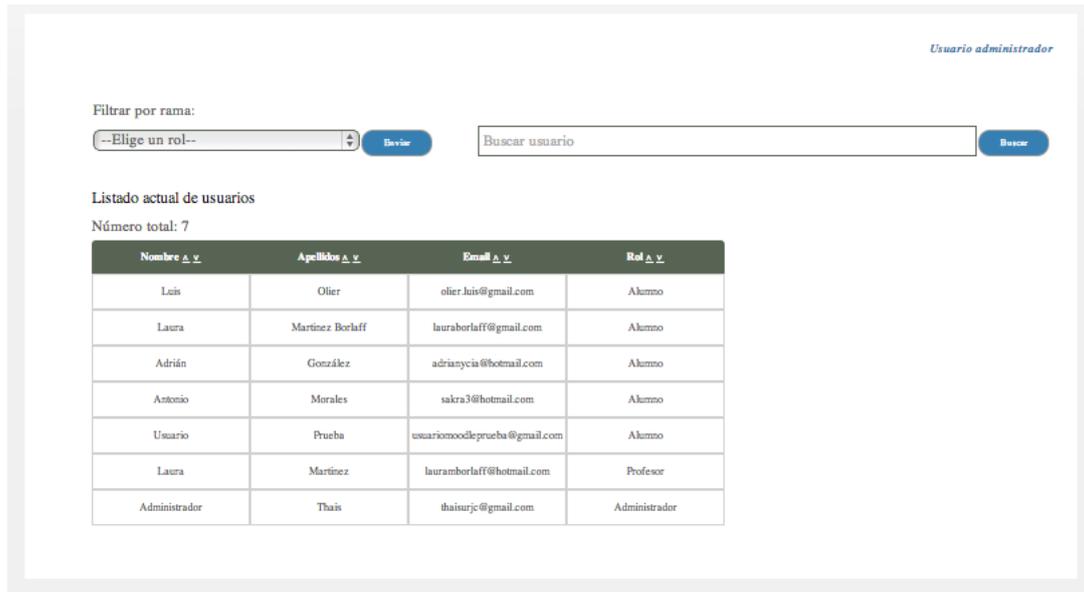


Figura A.21: Página de listado de usuarios de administrador en THAIS

A.3.8. Página de cambio de rol a usuarios

Esta página contiene un formulario donde introducir la dirección de correo del usuario que se quiera cambiar el rol y el nuevo rol que se desea que tenga de ahora en adelante [figura A.22].

Figura A.22: Página de cambio de rol a usuarios de administrador en THAIS

La dirección de correo del usuario debe ser la que éste este usando dentro de la plataforma. Este usuario recibirá un correo notificándole el cambio.

A.3.9. Página de cambio de contraseña a usuarios

Situada en Usuarios→Cambiar contraseña usuarios. En esta página deberemos introducir en el formulario que se muestra el correo electrónico del usuario que haya pedido el cambio de su clave dentro de la herramienta. Se creará una clave automática para este usuario y se le notificará vía email [figura A.23].



The screenshot shows a web form titled "Cambiar password usuario". Below the title, there is a label "Email del usuario:" followed by a text input field containing the placeholder text "Correo electrónico del usuario (se le enviará un mail con una password aleatoria)". At the bottom right of the form is a blue button labeled "Enviar".

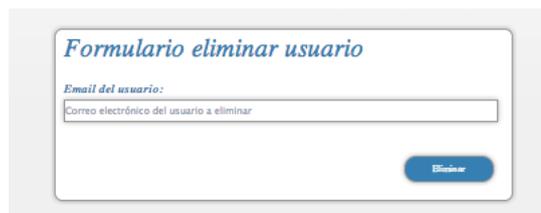
Figura A.23: Página de cambio de contraseña a usuarios de administrador en THAIS

A.3.10. Página de cambio de contraseña

Se encuentra en Usuarios→Cambiar contraseña. Este punto es exacto al redactado en alumno.

A.3.11. Página para eliminar usuarios

Dirigirse a Usuarios →Eliminar usuarios. Aparecerá un formulario donde introducir el correo de la cuenta de usuario a borrar [figura A.24].



The screenshot shows a web form titled "Formulario eliminar usuario". Below the title, there is a label "Email del usuario:" followed by a text input field containing the placeholder text "Correo electrónico del usuario a eliminar". At the bottom right of the form is a blue button labeled "Eliminar".

Figura A.24: Página para eliminar usuarios de administrador en THAIS

Apéndice B

Instalación de Moodle y tecnologías requeridas

Como se ha comentado en el texto, para este proyecto fin de carrera se ha elegido la versión de Moodle 2.2.4 (Build 20120706), versión actual en la ETSIT-URJC, Apache2 como servidor web, PHP 5.3.2 y MySQL. Aparte de todo esto, es necesario instalar phpmyadmin para manejar la base de datos de Moodle.

B.1. Preparación del entorno

Como trabajamos sobre el sistema operativo Ubuntu 12. Para preparar el entorno necesario para el correcto funcionamiento de Moodle, tenemos que instalar el software necesario que más adelante será configurado, si es necesario [28].

Vamos a empezar por instalar el servidor Apache2. Abrimos un terminal y ejecutamos los siguientes comandos: `sudo apt-get install apache2`, con este comando instalamos los paquetes de Apache2, y `sudo service apache2 start`, para iniciar el servicio [28] [29].

Tras estas dos órdenes, abrimos un navegador y escribimos la url: `http://localhost`. Tras cargar esta dirección aparecerá el mensaje "It Works" en la página [28].

A continuación instalamos la base de datos que vamos a usar para la plataforma, MySQL. Para ello debemos instalar los paquetes necesarios por medio del comando `sudo apt-get install mysql-server`, que nos pedirá a continuación que definamos la contraseña para el usuario root. El servidor, en resumen, es el que contiene todos los recursos como diccionario de datos o rutinas de usuario, y además donde residen las bases de datos con sus espacios de almacenamiento e indexación y sus motores de optimización para las consultas. También tenemos que instalar el cliente mediante el comando `sudo apt-get install mysql-client`. El cliente contiene los recursos de intérprete, herramientas para la administración de las tablas de datos y parseo de consultas [28] [29].

A continuación instalaremos el software para PHP. Escribimos en el terminal: `sudo apt-get install php5` y el administrador `sudo apt-get install phpmyadmin`. Tras estas órdenes nos pedirá ingresar al contraseña del usuario root de MySQL y también escoger nuestra clave para phpmyadmin [28] [29].

B.1.1. Comprobando que el entorno es el adecuado

Tras la instalación de Apache2, verificamos que estaba ejecutándose adecuadamente, cuando cargamos la dirección `http://localhost` y nos devolvió el mensaje de "It works". Ahora comprobamos si el software de PHP instalado funciona como queremos [28] [29].

La carpeta que se encuentra en el siguiente path `/var/www` es donde se deberán colocar todos los documentos que queramos que sean publicados por Apache. Si en ella editamos un fichero en PHP en el que llamamos a la función `phpinfo()`, de tal forma:

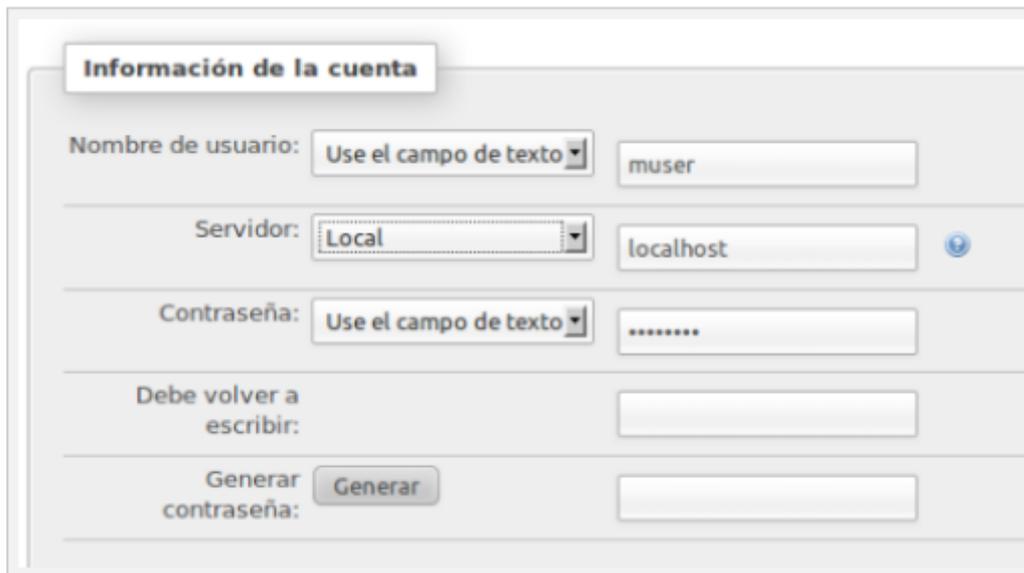
```
<?php
phpinfo();
?>
```

Llamando a este fichero, por ejemplo, `info.php` de tal forma que su path sea `/var/www/info.php`. Está función simplemente muestra en pantalla el estado de PHP y del servidor Apache, así como información sobre varios de sus componentes relacionados. Para ello debemos escribir la dirección `http://localhost/info.php` en nuestro navegador [28] [29].

B.2. Configuración de la base de datos para Moodle

Tras comprobar que el servidor y PHP funcionan como queremos, el siguiente paso a realizar es la creación de la base de datos que usará Moodle. Para ello, usamos el navegador para cargar la dirección `http://localhost/phpmyadmin` e ingresamos la clave requerida. Mediante este administrador vamos a tener acceso a la base de datos de MySQL y nos va a permitir crear de una forma fácil la base de datos. Desde la página a la que hemos accedido tras introducir la clave de usuario root, nos dirigimos a la pestaña que dice "Base de datos", y, de allí hacemos click en "Crear base de datos". Introducimos como nombre "Moodle" y seleccionamos el cotejamiento "utf8-spanish-ci" y hacemos click en el botón de "Crear", que a continuación, nos debería notificar que se ha creado. Aparte de esto, Moodle requiere tener creado un usuario para la base de datos del LMS, y por ello vamos a crear un usuario con privilegios en la base de datos. En la misma sesión que hemos iniciado en PHPMYADMIN, seleccionamos nuestra base de datos ya creada, llamada Moodle, de la lista de bases de datos que hay situada a la derecha de la página de inicio de PHPMYADMIN. Entre las opciones, selec-

cionamos "Privilegios" en el menú izquierdo y hacemos click en "Agregar un nuevo usuario". Tendremos que rellenar unos campos con los siguientes datos en B.1 [28].



The image shows a web form titled "Información de la cuenta" (Account Information). It contains several input fields and a button:

- Nombre de usuario:** A dropdown menu with the text "Use el campo de texto" and a text input field containing "muser".
- Servidor:** A dropdown menu with the text "Local" and a text input field containing "localhost".
- Contraseña:** A dropdown menu with the text "Use el campo de texto" and a text input field with masked characters ".....".
- Debe volver a escribir:** A text input field for password confirmation.
- Generar contraseña:** A button labeled "Generar" and a text input field for the generated password.

Figura B.1: Información de la cuenta de usuario para Moodle. Fuente [28]

Y a continuación seleccionar los privilegios, de tal modo que B.2. Y una vez hecho esto hacer click en continuar.

B.2.1. Creación de la carpeta para los datos de Moodle

Moodle necesita una carpeta donde guardar todos los archivos que vaya creando o que sean subidos a él. Ésta carpeta que llamaremos "Moodledata", la vamos a crear en la carpeta de sistema `/var/`, y para ello basta con el ejecutar el comando `sudo mkdir /var/moodledata` en una ventana del terminal. Como la carpeta que acabamos de crear está contenida en otra carpeta del sistema, ésta hereda los permisos de la carpeta contenedora, los cuales excluyen los permisos de escritura y tenemos que dárle los permisos solicitados por Moodle, de la forma `sudo chmod 777 /var/moodledata` [28].

A continuación debemos instalar el módulo: `sudo apt-get install php5-curl` [28] [29].

B.3. El paquete Moodle

Una vez realizados todos los pasos anteriores, descargamos el paquete de Moodle, el cual, se puede adquirir desde la página oficial de Moodle en el apartado de descargas en la dirección: <http://download.moodle.org/>. Una vez descargado, se procederá a su descompresión y se deberá guardar la carpeta obtenida tras proceso en el directorio de nuestro servidor, de tal forma



Figura B.2: Privilegios del usuario para Moodle Fuente [28]

que el path de la carpeta que contiene el paquete Moodle descargado sea `/var/www/moodle` [28].

Una vez almacenada la carpeta en el sitio descrito, abrimos el navegador y escribimos la dirección `http://localhost/moodle` y seguiremos las opciones que aparezcan por pantalla, pulsando en "Siguiete" [28].

Finalizando la instalación de Moodle, tenemos que crear el fichero, por medio de `sudo gedit /var/www/moodle/config.php` y lo editarlo mediante las siguientes líneas o con las líneas de código que el instalador nos devuelve:

```
<?php // Moodle configuration file

unset($CFG);
global $CFG;
$CFG = new stdClass();

$CFG->dbtype = mysqli;
$CFG->dblibrary = native;
$CFG->dbhost = localhost;
$CFG->dbname = moodle;
$CFG->dbuser = mooduser;
```

```
$CFG->dbpass = *****;
$CFG->prefix = mdl_;
$CFG->dboptions = array (
dbpersist => 0,
dbsocket => 0,
);

$CFG->wwwroot = http://localhost/moodle;
$CFG->dataroot = /var/moodledata;
$CFG->admin = *****;

$CFG->directorypermissions = 0777;

$CFG->passwordsaltmain = *****;

require_once(dirname(__FILE__) . /lib/setup.php);

// There is no php closing tag in this file,
// it is intentional because it prevents trailing whitespace problems!
```

Los campos en los que aparece el símbolo "*" deben de ser completados con las contraseñas elegidas durante todo el proceso. Guardamos el fichero y la instalación está completada.

Escribiendo la url `http://localhost/moodle` podremos comenzar nuestro trabajo con Moodle [28].

B.4. Problemas encontrados

Durante la instalación de Moodle, me encontré los siguientes problemas.

B.4.1. MySQL-SERVER

Creí que no instalé adecuadamente MySQL, debido a que no funcionaba de la manera deseada, y opté por borrar el paquete y volver a instalarlo mediante los siguientes pasos en una consola:

1. Escribir el comando `sudo bash` e introducir la contraseña root.
2. Ejecutar la orden `apt-get remove --purge mysql-server` para borrar y desinstalar los ficheros

3. Escribir por terminal `apt-get update`
4. Y finalmente el comando citado en la explicación de instalación `apt-get install -y mysql-server`

B.4.2. PHPMYADMIN

Después de instalar el servidor Apache2, MySQL-server, PHP y PHPMYADMIN, cuando traté de cargar la página de PHPMYADMIN obtuve el siguiente error:

```
The mysqli extension is missing. Please check your PHP configuration.  
<a href=Documentation.html#faqmysql  
target=documentation><img src=./themes/pmahomme/img/b_help.png  
width=11 height=11 alt=Documentation title=Documentation /></a>
```

La solución a él que encontré fue que tuve que comprobar que tenía bien instalado `mysql-server` y `php5-mysql`, y en caso de que no ejecutar el comando `sudo apt-get install php5-mysql` y reiniciar el servidor Apache antes de cargar la página de PHPMYADMIN, con el comando: `sudo /etc/init.d/apache2 restart`.

Bibliografía

- [1] GALARRAGA, SOPHIA. "*Tipos de web*". Junio 2011. [Julio 2012]. Disponible en la web: <http://www.slideshare.net/sgalarra2009/tipos-de-web-8293937>
- [2] "*Tipos de sitios web*". Mayo 2007. Mayo 2007. [Septiembre 2012]. Disponible en la web: <http://www.lawebera.es/manuales/primeros-pasos/como-empezar/tipos-de-webs.php> Publicado por *Pimienta y Amanda*
- [3] WIKIPEDIA. "*Web 1.0*". Octubre 2008. Agosto 2012. [Julio 2012]. Disponible en la web: <http://es.wikipedia.org/wiki/Web-1-0>
- [4] WIKIPEDIA. "*Web 2.0*". Octubre 2005. Septiembre 2012. [Julio 2012]. Disponible en la web: <http://es.wikipedia.org/wiki/Web-2-0>
- [5] WIKIPEDIA. "*Web semántica*". Octubre 2005. Septiembre 2012. [Julio 2012]. Disponible en la web: <http://es.wikipedia.org/wiki/Web-semántica>
- [6] W3C ESPAÑA. "*Guía breve de Web Semántica*". [Julio 2012]. Disponible en la web: <http://www.w3c.es/Divulgacion/GuiasBreves/WebSemantica>
- [7] FACTORÍA DE INTERNET. "*Guía breve de Web Semántica*". 2003. 2011. [Julio 2012]. Disponible en la web: <http://www.webtaller.com/maletin/articulos/acerca-web-3.php>
- [8] SOCIAL MEDIA MARKETING. "*Breve historia de las redes sociales*". Enero 2011. [Julio 2012]. Disponible en la web: <http://www.marketingdirecto.com/actualidad/social-media-marketing/breve-historia-de-las-redes-sociales/>
- [9] BENNETT, SHEA. "*The History of Social Media (1978-2012)*". Febrero 2012. [Julio 2012]. Disponible en la web: <http://www.mediabistro.com/alltwitter/social-media-history-b18776>
- [10] MÚJICA, JORGE ENRIQUE. "*Redes sociales: historia, oportunidades y retos*". Marzo 2010. [Julio 2012]. Disponible en la web: <http://www.forumlibertas.com/frontend/forumlibertas/noticia.php?id-noticia=16428>

- [11] LINKEDIN. "*Company Background*". [Julio 2012]. Disponible en la web: <http://press.linkedin.com/about>
- [12] LINKEDIN. "*LinkedIn Founders*". [Julio 2012]. Disponible en la web: <http://press.linkedin.com/founders>
- [13] MOODLE. "*About Moodle*". Mayo 2005. April 2012. [Julio 2012]. Disponible en la web: <http://docs.moodle.org/23/en/About-Moodle>
- [14] MOODLE. "*Moodle Statistics*". [Julio 2012]. Disponible en la web: <http://moodle.org/stats/>
- [15] ÁNGEL CORROCHANO, ALEXANDER. Directores: Gamez Martín, José Antonio y Parreño Torres, Francisco. "*Novedades para la docencia en Moodle 2.0*". Septiembre 2010. Proyecto Fin de Carrera. Universidad de Castilla-La Mancha. Escuela Superior de Ingeniería Informática. Departamento de Sistemas Informáticos. Departamento de Matemática Aplicada. ISBN: 978-1-4467-2953-3
- [16] GONZÁLEZ DE FELIPE, ANA TERESA. "*Guía de apoyo para el uso de Moodle 1.9.4 Usuario Administrador*". Universidad de Oviedo. Escuela Universitaria de Ingeniería Técnica de Informática de Oviedo (EUITIO).
- [17] "*Actividad 5 - Web 1.0 vs Web 2.0*". Junio 2011. [Agosto 2012]. Disponible en la web: <http://comamosbien.wordpress.com/2011/06/19/actividad-5-web-1-0-vs-web-2-0/> Publicado por *comamosbien*
- [18] BELLEDI, FABIÁN. "*Web 1.0, Web 2.0 y Web 3.0*". Marzo 2009. [Agosto 2012]. Disponible en la web: <http://fabianbelledi.wordpress.com/2012/03/09/web-1-0-web-2-0-y-web-3-0/>
- [19] BENNETT, SHEA. "*A Short History Of Social Media (1978-2011)*". Agosto 2011. [Agosto 2012]. Disponible en la web: <http://www.mediabistro.com/alltwitter/history-social-media-b12770>
- [20] BOYD, DANA M.. School of Information. University of California-Berkeley. ELLISON, NICOLE B.. Department of Telecommunication, Information Studies, and Media. Michigan State University. "*Social Network Sites: Definition, History, and Scholarship*". Journal of Computer-Mediated Communication, 13(1), article 11. 2007. [Agosto 2012]. Disponible en la web: <http://jcmc.indiana.edu/vol13/issue1/boyd.ellison.html>
- [21] MAUSSON, ANTOINE. "*The Digital Natives*". Karlskrona (Suecia). Noviembre 2011. Presentacion en formato PDF para la asignatura *Marketing of high-tech products, services and experiences*. Blekinge Tekniska Högskola (BTH).

- [22] WIKIPEDIA. "*Tim Berners-Lee*". Agosto 2001. Septiembre 2012. [Septiembre 2012]. Disponible en la web: <http://en.wikipedia.org/wiki/Tim-Berners-Lee>
- [23] GARRIDO CURBEIRA, JUAN CARLOS. "*Facebook supera ya los 900 millones de usuarios*". Abril 2012. [Septiembre 2012]. Disponible en la web: <http://dtecn.com/facebook-supera-ya-los-900-millones-de-usuarios/>
- [24] "*Acerca de Moodle*". Febrero 2006. Agosto 2007. [Noviembre 2012]. Disponible en la web: <http://docs.moodle.org/all/es/Acerca-de-Moodle>
- [25] "*Moodle*". Noviembre 2004. Noviembre 2012. [Noviembre 2012]. Disponible en la web: <http://es.wikipedia.org/wiki/Moodle>
- [26] GIROUX, ALEXANDRA. "*VLE, CMS, LMS OR LCMS*". 2011. [Noviembre 2012]. Disponible en la web: <http://www.alexandragiroux.net/vle-cms-lms-or-lcms/>
- [27] ANCA GARCÍA, MARÍA ASUNCIÓN. "*Fundamentos teóricos del Constructivismo pedagógico*". Mayo 1997. [Noviembre 2012]. Disponible en la web: <http://www.monografias.com/trabajos27/constructivismo-pedagogico/constructivismo-pedagogico.shtml>
- [28] "*How to: Instalar Moodle en Ubuntu 11.04*". Marzo 2012. [Noviembre 2012]. Disponible en la web: <http://nlinux.wordpress.com/2012/03/19/how-to-instalar-moodle-en-ubuntu-11-04/>
- [29] "*Step-by-step Installation Guide for Ubuntu*". September 2012. [Noviembre 2012]. Disponible en la web: <http://docs.moodle.org/23/en/Step-by-step-Installation-Guide-for-Ubuntu>
- [30] "*Características*". Febrero 2006. Marzo 2008. [Noviembre 2012]. Disponible en la web: <http://docs.moodle.org/all/es/Caracteristicas>
- [31] "*Ventajas y desventajas de Moodle*". Septiembre 2011. [Noviembre 2012]. Disponible en la web: <http://www.fix-si.com/blog/2011/09/ventajas-y-desventajas-de-moodle/>
- [32] "*Nvidia, Continuum team up to sling Python at GPU coprocessors*". Marzo 2013. [Mayo 2013]. Disponible en la web: <http://www.theregister.co.uk/2013/03/18/nvidia-continuum-python-on-cuda-gpu/>
- [33] "*Top 20 programming languages of hacker news readers January 2013*". Abril 2013. [Mayo 2013]. Disponible en la web: <http://devopsangle.com/2013/04/02/top-20-programming-languages-of-hacker-news-readers-january-2013/>

- [34] "12 must-know advantages of PHP". Junio 2012. [Mayo 2013]. Disponible en la web: <http://vandelaydesign.com/blog/web-development/advantages-of-php/>
- [35] "Disadvantages of web development using PHP". Junio 2009. [Mayo 2013]. Disponible en la web: <http://www.itbusinessedge.com/cm/blogs/mah/disadvantages-of-web-development-using-php/?cs=33397>
- [36] "Beginner's Guide to: building HTML5/CSS3 webpages". [Mayo 2013]. Disponible en la web: <http://www.hongkiat.com/blog/building-html5-css-webpages/>
- [37] MÁRQUEZ GÓMEZ, JOSÉ JORGE. "Arquitectura MVC. Visión general". [Mayo 2013]. Disponible en la web: <http://jorge.queideas.com/wp-content/uploads/2011/11/Arquitectura-MVC.pdf>
- [38] "Foro sobre hojas de estilo CSS". Septiembre 2006 [Mayo 2013]. Disponible en la web: <http://www.webestilo.com/foros/mensaje.phtml?foro=16&msg=47073>
- [39] "Practical CSS3 tables with rounded corners". Octubre 2011 [Mayo 2013]. Disponible en la web: <http://www.red-team-design.com/practical-css3-tables-with-rounded-corners>
- [40] "Alterar colores por fila en una tabla y resaltar fila al pasar el mouse ". [Mayo 2013]. Disponible en la web: <http://www.summarg.com/2010/alternar-colores-por-fila-en-una-tabla-y-resaltar-fila-al-pasar-el-mouse/>
- [41] "HTML5 LED scroller using a Canvas Element ". Julio 2012. [Mayo 2013]. Disponible en la web: <http://geeksretreat.wordpress.com/2012/07/07/html5-led-scroller-using-a-canvas-element/>
- [42] "Manual de PHP ". 2001. Mayo 2013. [Mayo 2013]. Disponible en la web: <http://www.php.net/manual/es/index.php>
- [43] BUCHANAN, BEN. "Estilos de las tablas". Septiembre 2008. [Mayo 2013]. Disponible en la web: <http://mosaic.uoc.edu/ac/le/es/m6/ud7/index.html>
- [44] WEAKLEY, RUSS Y GEORGE, JAMES. "Styling forms with CSS". [Mayo 2013]. Disponible en la web: <https://learnable.com/courses/styling-forms-with-css-220/preview/lesson/1/step/4> y <http://www.sitepoint.com/author/jgeorge/>
- [45] "25 Css form templates and input styles". [Mayo 2013]. Disponible en la web: <http://www.qubesys.com/25-css-form-templates-and-input-styles/>
- [46] "Apache Tutorial: .htaccess". [Mayo 2013]. Disponible en la web: <http://httpd.apache.org/docs/2.0/howto/htaccess.html#when>

- [47] "*Package Information: Mail*". [Mayo 2013]. Disponible en la web: <http://pear.php.net/package/Mail/redirected>
- [48] "*Authentication: LinkedInAuth*" y "*Blocks: LinkedIn*". [Mayo 2013]. Disponible en la web: https://moodle.org/plugins/view.php?plugin=auth_linkedin y https://moodle.org/plugins/view.php?plugin=block_linkedin respectivamente.
- [49] "*Customisable user profiles*". [Mayo 2013]. Disponible en la web: http://docs.moodle.org/dev/Customisable_user_profiles
- [50] "*User profile fields: CertificatesWall profile*". [Mayo 2013]. Disponible en la web: https://moodle.org/plugins/view.php?plugin=profilefield_certificateswall
- [51] "*Profile field enrolment*". [Mayo 2013]. Disponible en la web: http://docs.moodle.org/22/en/Profile_field_enrolment
- [52] "*Mahoodle*". [Mayo 2013]. Disponible en la web: <http://manual.mahara.org/en/1.5/mahoodle/mahoodle.html>
- [53] "*II edición del curso Desarrollo web con HTML5 y CSS3*". [Mayo 2013]. Disponible en la web: <http://www.um.es/aulafcd/html5/>