



II JORNADAS DE INNOVACIÓN DOCENTE, TECNOLOGÍAS DE LA INFORMACIÓN Y DE LA COMUNICACIÓN E INVESTIGACIÓN EDUCATIVA EN LA UNIVERSIDAD DE ZARAGOZA 2008

Gestión Automatizada de los Proyectos Finales de Carrera de la Universidad de Zaragoza

D^a Piedad Garrido

Asociado TC/EUPT

D. Francisco J. Martínez

TEU/EUPT

Dr. D. Jesús Tramullas

TU/Facultad de Filosofía y Letras

Dr. D. Miguel A. Estebán

TU/Facultad de Filosofía y Letras

D. Fernando Naranjo

Asociado TP y PAS del Servicio de Informática/EUPT

D^a Lucía Calvo

Estudiante de Ingeniera Técnica en Informática de Gestión/EUPT

Síntesis:

Esta contribución versa sobre la descripción de un proyecto de software libre desarrollado en la propia Universidad de Zaragoza (UZ) para la gestión interna de los Trabajos Finales de Carrera (TFCs). Este proyecto fue respaldado por la línea de trabajo de utilización de herramientas TIC especializadas en metodologías activas del PESUZ-2006 y se fundamenta en la construcción de un entorno de gestión de los TFCs “ad-hoc” haciendo uso de workflow. Para su desarrollo se montó un equipo de trabajo multidisciplinar e interdepartamental (PAS, PDI y alumnado de la UZ), cuyos miembros se encuentran trabajando de forma activa en la adaptación de sus asignaturas al EEES, y entornos semipresenciales. Consiguiendo finalmente una herramienta de gestión de gran utilidad tanto para al profesorado como para el alumnado, puesto que dispondrán en un único espacio de trabajo de toda la documentación relativa al desarrollo de un TFC y una biblioteca digital con TFCs de años anteriores con la propiedad intelectual cubierta por una licencia Creative Commons.

Palabras clave

Software libre, Trabajo Final de Carrera, accesibilidad y licencia

1. INTRODUCCIÓN

La idea principal del presente trabajo surgió de la necesidad de gestionar el flujo de trabajo (*workflow*¹) generado por la dirección de los Trabajos Finales de Carrera (TFCs) supervisados de forma continuada, con el objetivo de poder llevar a cabo un seguimiento de todo el ciclo de vida del trabajo del alumno en esta asignatura, hasta la hora de acordar una fecha de exposición concreta. Este seguimiento tiene una estructura ordenada, definida y hay que asegurar la ejecución de todos los pasos necesarios para la consecución del mismo; es decir, llegar a tener un control completo de la evolución del Trabajo Final de Carrera por parte del tutor y/o del profesor.

Primero, es necesario conocer el estado académico en el que se encuentra el alumno, por lo que se le solicita información específica de su expediente (*nota media, asignaturas pendientes, etc*) con el objetivo de poder realizar una asignación adecuada. Una vez hecha la asignación de un TFC, el software desarrollado, tal y como se va a poder observar más adelante, manipula datos reales, lo que incrementa su usabilidad de cara a la figura del administrador del sistema y el posterior mantenimiento del mismo. Cuando el TFC haya sido presentado y evaluado, su permanencia en el centro académico es fundamental para que pueda ser consultado por futuros alumnos, por lo que se ha integrado una herramienta para que los proyectitas puedan subir los archivos de sus correspondientes TFCs a un servidor, partiendo del supuesto de que la propiedad intelectual de los TFCs está protegida bajo la licencia Creative Commons. Los ficheros y la estructura de almacenamiento de los mismos, es compatible con otra aplicación anterior, también desarrollada en la EUPT, en la que se implementó una biblioteca digital, cuya finalidad es almacenar en formato electrónico todas las memorias de los TFCs leídos en el centro académico para que estén a disposición del resto del alumnado y personal del centro en cualquier lugar y en cualquier momento.

La ponencia cuenta con los siguientes apartados: una introducción donde se comenta cómo surgió esta idea, los objetivos que se han cubierto, un apartado en el que se muestra que no hay en el mercado aplicaciones que sean capaces de cubrir las necesidades planteadas, un apartado de descripción del sistema de información, donde se detallan aspectos tales como la arquitectura de la aplicación, las tecnologías utilizadas, el diseño de la base de datos, estándares y licencias utilizadas y aspectos de accesibilidad y usabilidad. Para finalizar con un apartado de conclusiones en el que el personal involucrado en esta tarea refleja sus impresiones sobre el proyecto realizado.

2. OBJETIVOS

Los objetivos planteados y que finalmente han sido alcanzados son los que se detallan a continuación:

- Informatizar la gestión de los TFCs mediante flujos de trabajo
- Almacenar y clasificar toda la información relacionada con cada uno de los TFCs
- Seguimiento y registro del proceso de desarrollo de los TFCs

¹ El término hace referencia a la automatización de un proceso llevado a cabo para la consecución de una tarea o trabajo predeterminado

- Contener toda la reglamentación vigente para la realización de un TFC, y poder proporcionársela al alumno/a
- Realizar un seguimiento detallado de la realización de un TFC, desde la primera entrevista con el alumno, pasando por el control de las distintas reuniones de trabajo hasta la obtención de la información del resultado del proceso, como por ejemplo día y nota de la exposición
- Servir como punto de información sobre los TFCs (en la *extranet*), tanto la normativa, como los requisitos y documentos necesarios; así como algunos consejos útiles y enlaces para la realización del mismo
- Se dispondrá de un acceso de permiso total (figura del administrador) para poder gestionar la información de la base de datos para que sea lo más real posible. Desde aquí se podrán actualizar todos los cambios relacionados con el profesorado, departamentos, áreas y comisiones del centro²
- Realizar una aplicación web que cumpla con los requisitos de usabilidad y accesibilidad de cara al usuario y que sea fácil de mantener y actualizar, por parte del administrador

3. ESTADO DEL ARTE

El estado del arte se centró en la recogida y evaluación de productos de empresas especializadas en workflow y productos workflow de software libre. De cada una de estas aplicaciones, se analizaron tanto los contenidos como las tecnologías utilizadas.

1. En cuanto a tecnologías se tuvieron en cuenta:

- Lenguaje: lenguaje en el que se ha implementado el producto
- SGBD: sistema gestor de base de datos que soporta
- SO: sistema operativo.
- Software Libre o propietario

2. En cuanto a contenidos se analizaron:

- Tipo: tipo de workflow
- Ámbito de Aplicación. Si se mueve en un entorno empresarial, gubernamental o educativo

Los productos analizados fueron:

- Staffware de TIBCO
- COSA Workflow de COSA SOLUTIONS
- @enterprise de Groiis
- XFlow
- Wftk
- OpenWFE

² En este caso en particular, los datos proceden de la Escuela Universitaria Politécnica de Teruel (EUPT)

La conclusión a la que se llegó con este estudio, es que no se ha encontrado ningún producto válido para gestionar los TFCs.

Una de las principales razones es que los productos workflow están orientados hacia las empresas. La definición de los flujos de trabajo no es tan configurable como para adaptarlo al entorno académico. Además lo que se entiende hoy como producto workflow es muy heterogéneo. Las diferencias en contenidos y funcionalidades son grandes, y la definición de flujo de trabajo para cada producto es totalmente distinta.

Aunque se hayan analizado productos de empresas especializadas de workflow, se ha observado como cada vez las empresas relacionadas con las tecnologías de la información realizan su propio paquete workflow, lo cual hace notar que cada producto es específico para unas situaciones concretas. Del soporte de los Sistemas Gestores de Base de Datos (SGBD), se puede decir que la gran mayoría sólo aceptan SGBD de software propietario. La tecnología a la hora de implementar estos productos es, en la gran mayoría de los casos, Java. En cuanto a los sistemas operativos, todos admiten Windows. Aunque también se empiezan a ver productos que funcionan bajo plataformas Unix y/o Linux.

Y en lo que respecta al ámbito de aplicación de estos productos, está orientado hacia el ámbito privado o empresarial. También cabe destacar el uso que hacen del mismo las administraciones públicas, siendo notorio que cada administración usa un producto workflow distinto. Existen gran cantidad de productos workflow en el mercado, tanto de software libre como de software propietario. Sin embargo, las funcionalidades, contenidos y aplicaciones de los mismos son muy diversas.

Por lo que, a falta de un producto concreto que se pueda adaptar a las necesidades detalladas en los objetivos se decidió desarrollar un workflow “ad-hoc” para la gestión de TFCs de la EUPT.

4. DESCRIPCIÓN DEL SISTEMA DE INFORMACIÓN

En este apartado se va a llevar a cabo una descripción del software desarrollado. Se comenzará con una muestra gráfica de la arquitectura del sistema de información (*figura 1*), se describirán las tecnologías y herramientas utilizadas en su implementación, el diseño de la base de datos que incorpora, los estándares y licencias involucrados que determinarán la licencia final de la aplicación software desarrollada, para terminar comentando los aspectos de accesibilidad y usabilidad tenidos en cuenta a lo largo de su diseño.

4.1 Arquitectura del Sistema

Destacar que la aplicación en el cliente ha sido probada en los siguientes navegadores: Internet Explorer, Mozilla Firefox y Ópera.

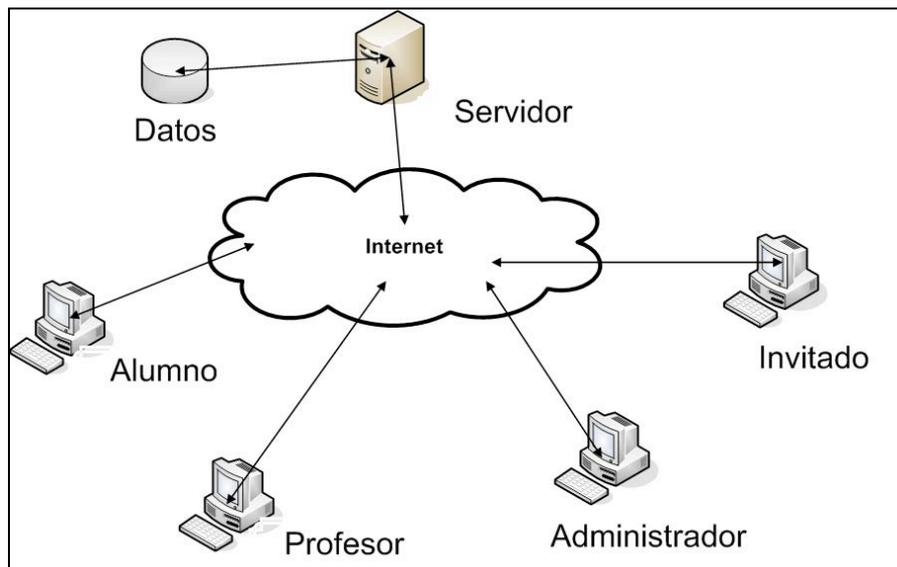


Figura 1 Arquitectura del Sistema de Información

4.2 Tecnologías y Herramientas Utilizadas

En cuanto a las tecnologías y herramientas utilizadas en su implementación, son las siguientes:

1. La interfaz de aplicación de programa Lucene: una API (Application Program Interface) de desarrollo para indexación y búsqueda, multiplataforma, de código abierto y que se distribuye bajo la Apache Software License [2,6 y 13].
2. Lenguaje de programación Java: lenguaje de programación multiplataforma, orientado a objetos y de código abierto [9].
3. El entorno de desarrollo integrado NetBeans: una plataforma para el desarrollo de aplicaciones usando el lenguaje de programación anteriormente mencionado [14].
4. Tomcat: Servidor Web con soporte para las tecnologías Java Server Pages y Java Servlets. Su código fuente es de libre acceso y sus ficheros binarios se encuentran al amparo de la Apache Software Licence [10, 15].
5. CSS (*Cascading Style Sheets*): Las hojas de estilo en Cascada son un lenguaje formal usado para definir la presentación de un documento estructurado escrito con HTML ó XML [1].
6. Lenguaje HTML (**H**yper**T**ext **M**arkup **L**anguage): Lenguaje de marcado hipertextual en el que se programan las páginas web, y que es interpretado posteriormente por los navegadores web (browsers). La especificación de su última versión se puede consultar en la siguiente URL [8]
7. DerEditor: Aplicación de código abierto, multiplataforma y orientada al Diseño de Bases de Datos. Destacar que esta aplicación fue desarrollada por un alumno de la EUPT como TFC y está disponible en [3]

8. FireBird: Sistema Gestor de Base de Datos Relacional (SGBDR) de código abierto, multiplataforma bajo una licencia similar a la MPL (Mozilla Public License) [12].

4.3 Diseño de la Base de Datos

El Diseño de la Base de Datos ha sido una de las componentes clave del sistema. El esquema conceptual final, después de haber pasado por los siguientes estadios de diseño: conceptual, lógico, físico, y normalización, es el que se muestra en la siguiente figura:

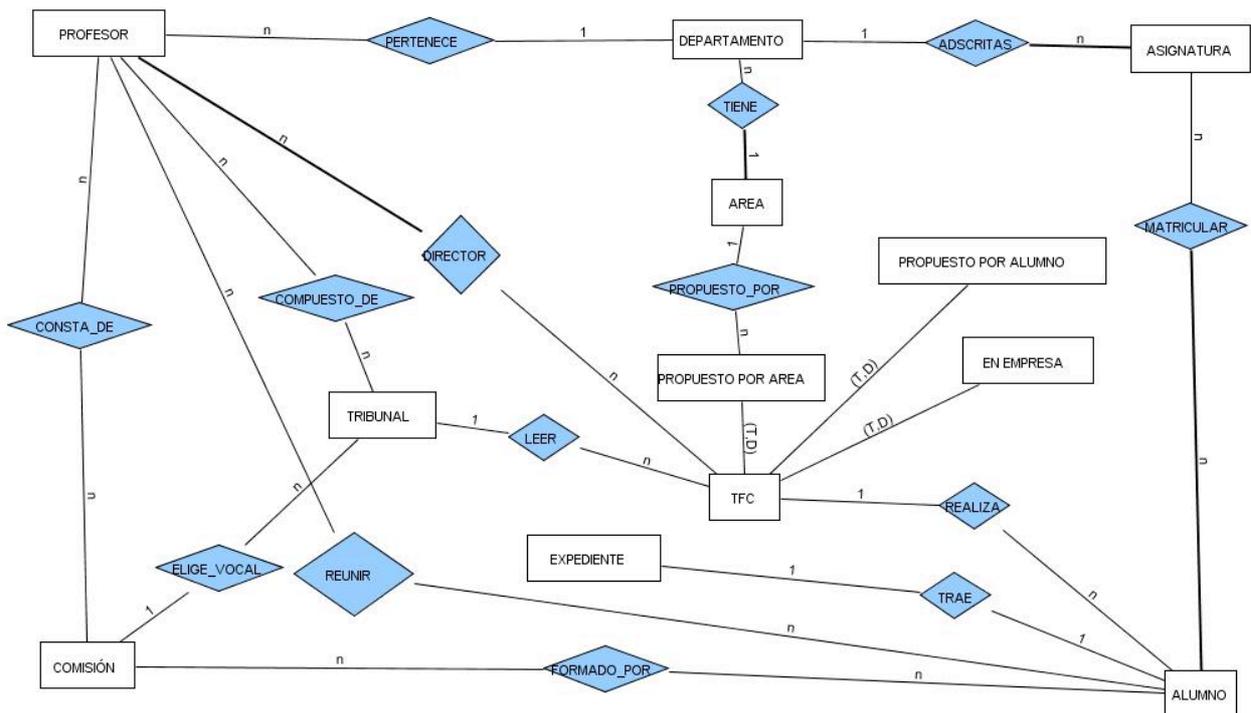


Figura 2 Entidad Relación (E-R)

Como se puede observar, las entidades que forman parte del sistema son: profesor, alumno, asignatura, departamento, comisión, tribunal, expediente, área y Trabajo Final de Carrera (TFC). El comportamiento del sistema de información lo definen las relaciones, que son las estructuras representadas con un rombo azulado y sus cardinalidades [11].

4.4 Estándares y Licencias Utilizadas

Los estándares utilizados han sido propuestos por el W3C (World Wide Consortium), el IEEE (Institute of Electrical and Electronics Engineers) y la ISO (Internacional Standard Organization). Las dos primeras organizaciones desarrollan estándares y normas referentes a ordenadores, comunicaciones y desarrollo y expansión de la Web, mientras que la tercera es la organización internacional de normalización por excelencia.

Para la aplicación desarrollada se ha hecho uso de: CSS, HTML, IEEE 830-1198 y las normas ISO 690: 1987 e ISO 690-2:1997 [4]. Tanto las CSS como el HTML forman parte de la implementación software. En cuanto a la IEEE 830-1998-Recommended Practice for Software Requirements Specifications, comentar que con esta norma se ha llevado a cabo la Especificación de Requisitos Software (ERS) de la aplicación. Esta especificación afecta a la etapa de desarrollo del diseño y su objetivo es definir de manera clara y precisa tanto las funcionalidades como las posibles restricciones del Sistema de Información a construir. Dicho documento irá dirigido tanto a los miembros del equipo de desarrollo del sistema como al cliente del proyecto. Las normas ISO 690:1987 e ISO 690-2:1997, dan cobertura al apartado de referencias bibliográficas de todo tipo de fuentes de información que ha sido necesario consultar para el desarrollo del proyecto.

En cuanto a las licencias involucradas, se va a distinguir entre licencias software y licencias de contenido, puesto que el proyecto dispondrá de estos dos tipos de licencia. Por lo que, en primer lugar se va a analizar la licencia software final del producto desarrollado para este proyecto.

Se tienen que analizar las licencias de los distintos tipos de tecnologías y herramientas involucradas en su desarrollo:

<i>Tecnología</i>	<i>Licencia</i>
Lucene	<i>Apache Software Licence</i> Libre y abierta, con patentes.
Netbeans	<i>Sun Public Licence</i>
Firebird	<i>Licencia similar a la MPL (Mozilla Public Licence)</i>
DEReditor	Licencia GNU GPL
TAW	Licencia GNU

Tabla 1. Licencia software de la tecnologías y herramientas utilizadas

Una vez conocidos estos datos, se puede pasar a decidir la licencia que se le otorgará al software desarrollado. En este caso, el equipo de desarrollo se ha declinado por una licencia **GNU GPL**, puesto que su propósito es declarar que el software cubierto por esta licencia es software libre [5].

En lo que respecta a la licencia de contenidos, se ha elegido la licencia *Creative Commons Reconocimiento - No comercial-Compartir bajo la misma licencia* para proteger los derechos de autor de los documentos que contiene y que se sean creados y/o almacenados en el sitio web [5].



Figura 3 Pie de la Interfaz

4.5 Accesibilidad y Usabilidad

Para comprobar a qué nivel se cumple con las pautas de accesibilidad y usabilidad, se ha sometido la aplicación a diferentes tests heurísticos y on-line. De todos ellos se van a destacar los resultados obtenidos por el test de accesibilidad *TAW* (Test de Accesibilidad Web). Con esta aplicación se comprueba el nivel de accesibilidad alcanzado en el diseño y desarrollo del proyecto en cuestión.

Este test se basa, para la realización de los análisis, en las reglas desarrolladas por el *WAI* (Web Accessibility Initiative), perteneciente al W3C. Estas reglas, denominadas Pautas de Accesibilidad al Contenido Web 1.0, se componen de catorce pautas que definen los principios del diseño accesible. Cada una de estas pautas se divide en uno o varios puntos de verificación.

Los puntos de verificación explican cómo se aplica la pauta en el desarrollo de contenidos web. Cada punto de verificación pretende ser lo suficientemente específico, como para que cualquiera que revise una página o sitio pueda comprobar que dicho punto ha sido satisfecho.

Inicialmente se introduce una dirección URL (dirección web). El sistema analiza la página, basándose en las Pautas de Accesibilidad al Contenido Web 1.0, y genera un informe HTML basado en la página analizada con información sobre el resultado del análisis.



Figura 4. Programa TAW

Una vez realizado el test, el informe generado muestra que se han cumplido los tres niveles de prioridad: WAI-A, WAI-AA y WAI-AAA.

Informe Resumen

URL	TAW	Problemas						Informe
		1		2		3		
		P	M	P	M	P	M	
http://localhost:8084/TFCFLOW/	●	0	0	0	0	0	0	Informe

Figura 5. Informe resumen TAW

Por lo que se ha colocado el icono correspondiente en la aplicación web.

Hay que hacer notar que a pesar de que se ha alcanzado un gran nivel de accesibilidad, éste se podría mejorar.

También se han realizado las validaciones del W3C para las CSS y el HTML del sitio web, por lo que se pueden incluir en el pie de la interfaz los iconos correspondientes a esas validaciones, tal y como se muestra en la figura 6.



Figura 6 Pie de la interfaz

5. CONCLUSIONES

Tras la realización de este proyecto se ha llegado a una serie de conclusiones. Son las siguientes:

- La automatización de tareas, sobre todo si es personalizada como es el caso, es un proceso en el cual invirtiendo un poco de esfuerzo, se obtienen, grandes resultados.
- Destacar la importancia de “hacer las cosas bien” [7]³, esto es, cumplir los estándares y licencias, si bien al principio es una ardua tarea, una vez acostumbrados, es la mejor forma de trabajar. Y aporta una versión más consecuente del trabajo.
- A la hora del diseño de la página web, existían muchas posibilidades. Se ha optado por una interfaz sencilla y no muy recargada, para que pueda ser “accesible” para todos.
- La realización de un Trabajo Final de Carrera es una tarea costosa en tiempo y recursos tanto para el tutor como para el alumno que la realiza, donde la “organización” forma un papel importante en el proceso.

Finalmente, tanto el Personal Docente Investigador (PDI) a tiempo completo o tiempo parcial como el Personal de Área y Servicios (PAS) concluye este trabajo

³ Regla Federal Express ó 1-10-100 de la calidad del software

comentando que un elemento fundamental en su desarrollo ha sido la composición de un equipo multidisciplinar con la representación de todas las figuras universitarias: personal, profesorado y alumnado, puesto que viene a suplir la deficiencia presentada en desarrollos de tareas complejas, que por mucho que nos empeñemos es prácticamente imposible que puedan ser desarrolladas por una sola persona o por un equipo de personas con la misma base de conocimientos. También nos gustaría agradecer la ayuda proporcionada a través del Programa de Enseñanza Semipresencial de la Universidad de Zaragoza (PESUZ 2006), puesto que es un buen punto de partida para incentivar la promoción de este tipo de desarrollos, aunque no es suficiente para dar cobertura a todo lo que conllevan. Finalizar, animando al resto de miembros de la Universidad de Zaragoza a involucrarse en este tipo de proyectos, puesto que ha sido una experiencia muy fructífera, ya que se consigue conocer mejor a compañeros de otras áreas, una relación más estrecha con el alumnado y, se desarrollan herramientas que permiten automatizar procesos de gestión del día a día que nos permiten ahorrar tiempo y que como bien dijo Miguel de Cervantes Saavedra: “*Mientras se gana algo no se pierde nada*”.

Referencias bibliográficas

1. CSS (1999), [Guías Breves Hojas de Estilo](#) [2008]
2. Canales, Roberto (2003-2005), [Primeros pasos con Lucene](#), [2007]
3. DerEditor (2005), [Software de 4ª Generación para el Diseño de Base de Datos](#) [2008]
4. Fernández, César (2004), [Creación de sitios web mediante estándares](#) [2007]
5. Gómez Labrador, Ramón (2005), [Tipos de Licencias de software](#) [2007]
6. Hatcher, Erik (2004), *Lucene in action: a guide to the java search engine*, Greenwich, Manning
7. Hernández, J.F. (2003), *La Calidad del software y su medida*, Madrid, Cerasa.
8. HTML (1999), [Especificación HTML 4.01](#) [2008]
9. Java (1994), [Sitio web oficial](#) [2008]
10. JavaHispano (2006), [Tutorial Java Server Pages \(JSP\)](#) [2007]
11. Kevine, Henr (2001), [Conectividad de la base de datos de Java](#) [2007]
12. Firebird (2001), [Sitio web Oficial](#) [2008]
13. Lucene (2001), [Sitio web Oficial](#) [2008]
14. NetBeans (2000), [Sitio web oficial](#) [2008]
15. Tomcat (2001), [Sitio web oficial](#) [2008]
16. UC3M (Universidad Carlos III de Madrid), [Cómo citar Bibliografía](#) [2008]