



**MEMORIA DE LAS ACCIONES DESARROLLADAS  
PROYECTOS DE MEJORA DE LA CALIDAD DOCENTE  
VICERRECTORADO DE PLANIFICACIÓN Y CALIDAD  
IX CONVOCATORIA (2007-2008)**



❖ **DATOS IDENTIFICATIVOS:**

**Título del Proyecto**

**Desarrollo de Estándares y Herramienta Software para el Seguimiento, Control y Evaluación de Resultados de las Actividades Prácticas de los Alumnos Universitarios.**

**Resumen del desarrollo del Proyecto**

Uno de los ejes fundamentales del nuevo modelo EEES estriba en el cambio que debe realizarse en la enseñanza práctica a los alumnos Universitarios. La enseñanza debe pasar por enseñar al alumno a cómo aprender y cómo aplicar los conocimientos teóricos a casos prácticos que simulen situaciones reales con las cuales el alumno se enfrentará posteriormente en el mercado de trabajo. En esta actividad laboral con la que se enfrentará el alumno en el mercado de trabajo están implicadas diferentes actividades y responsabilidades para las cuales se deben establecer mecanismos de aprendizaje en el nuevo modelo de enseñanza.

El objetivo de este proyecto ha consistido la especificación análisis y desarrollo de una herramienta software que facilite al docente el seguimiento, control y la posterior evaluación de los resultados en la enseñanza práctica de la docencia bajo el modelo ECTS.

Para ello se ha llevado a cabo el desarrollo de estándares tanto procedimentales, como documentales para su obligado seguimiento por parte de los alumnos en el desarrollo de la docencia práctica, simulando, de esta forma, los protocolos con los que los titulados podrán encontrarse en las empresas, y se ha desarrollado una herramienta basada en Internet, mediante la cual se han establecido, para cada caso (asignatura, trabajo de curso, práctica de laboratorio, etc.) los procedimientos a seguir, los flujos de comunicación, los productos/resultados a desarrollar, crear o producir, los mecanismos de control, el esfuerzo invertido, las responsabilidades asignadas, etc., de forma que se ha perseguido el conseguir una serie de objetivos fundamentales: a) enseñar al alumno a aplicar los contenidos a casos prácticos reales, b) enseñar al alumno a desarrollar su actividad profesional bajo unos estándares y protocolos establecidos, c) incentivar el trabajo colaborativo entre los alumnos, la toma de decisiones y de responsabilidades, d) establecer un mecanismo de difusión de los productos (de cualquier tipo) desarrollados por los alumnos que sirva de base de datos y fuente de información para futuros alumnos, y e) aportar al profesor una herramienta que le facilite su tarea docente con un alcance que no puede ser obtenido de forma manual y le proporcione una ingente cantidad de información a partir de la cual pueda, además de evaluar el trabajo del alumno, mejorar su planificación docente futura bajo el modelo ECTS.

	<b>Nombre y apellidos</b>	<b>Código del Grupo Docente</b>
<b>Coordinador/a:</b>	<b>IRENE LUQUE RUIZ</b>	<b>047</b>

**Otros participantes:**

<b>MIGUEL ANGEL GÓMEZ NIETO</b>	<b>047</b>
<b>GONZALO CERRUELA GARCÍA</b>	<b>047</b>
<b>CARLOS MARTINEZ PEDRAJAS</b>	<b>GI-TIC110</b>
<b>JOSÉ CHECA CLAUDEL</b>	<b>GI-TIC110</b>

**Asignaturas afectadas**

**TODAS**

**Nombre de la asignatura**

**Área de Conocimiento**

**Titulación/es**

**TODAS**

**TODAS**

**TODAS**

## 1. Introducción justificativa del proyecto

El Espacio Europeo de Educación Superior es un proyecto, impulsado por la Unión Europea para armonizar los sistemas universitarios europeos de manera que todos ellos tengan una estructura homogénea de títulos de grado y postgrado, es decir, una misma valoración de la carga lectiva de los estudios, cursos, asignaturas, calificaciones y una estructura de titulaciones y formación continua fácilmente entendible por todos los estados miembros. La Declaración de Bolonia sienta las bases para la construcción de un "Espacio Europeo de Educación Superior", organizado conforme a ciertos principios (calidad, movilidad, diversidad, competitividad) y orientado hacia la consecución entre otros de dos objetivos estratégicos: el incremento del empleo en la Unión Europea y la conversión del sistema Europeo de Formación Superior en un polo de atracción para estudiantes y profesores de otras partes del mundo.

Las nuevas directivas de formación de los alumnos, que permitirán la implantación de estas medidas, pasarán por una revisión del propio proceso de enseñanza y aprendizaje, ya que en caso contrario estaríamos ante un proceso de cambio puramente estructural, sin entrar para nada en lo sustantivo de ese cambio, centrar las acciones en el alumno, en su aprendizaje y en el desarrollo de las competencias requeridas en cada caso.

Por tanto, podríamos decir que lo que da sentido a este cambio que se propugna es que se contempla *la enseñanza basada en el aprendizaje del alumno* y que será la actividad de éste la que marcará la guía del trabajo universitario. Y ese es, el eje que articula el cambio, la relevancia que hay que dar a la preparación profesional de los alumnos.

La formación práctica de los alumnos universitarios en general y de las titulaciones de informática, en especial, debe suponer no sólo el aprendizaje en el uso de técnicas, herramientas y software (lenguajes, sistemas operativos, comunicaciones, etc.), sino la formación integral en el proceso o actividad de creación, desarrollo o producción.

En el caso particular de las titulaciones de Informática, esta formación debe suponer el aprendizaje del alumno en el desarrollo de software de calidad y la "*simulación*" del proceso, tal y como éste se lleva a cabo en las empresas. Esta simulación supone que el alumno tenga que realizar sus trabajos prácticos teniendo en cuenta los siguientes aspectos, entre otros:

- Enfrentándose a la solución de un problema nuevo, planteado en la asignatura correspondiente.
- Aportando una solución software al mismo.
- Trabajando en grupos.
- Coordinando su trabajo con otros grupos.
- Ajustando su trabajo a estándares establecidos.
- Siguiendo una serie de restricciones establecidas por el sistema, gestor, o software existente.

El desarrollo de los trabajos prácticos bajo estas características dará lugar a una formación del alumno más ajustada a la situación que el alumno se encontrará en el mercado de trabajo.

Esta metodología en el desarrollo de las prácticas se está llevando a cabo por diferentes profesores en algunas asignaturas de la Titulación de Ingeniería Superior en Informática. Sin embargo, diferentes problemas se presentan actualmente en su puesta en marcha, entre los que cabe citar:

- Los alumnos no están acostumbrados a la realización de trabajos en grupo.
- Los alumnos no están acostumbrados a asumir responsabilidades que implique toma de decisiones respecto al trabajo propio o de otros alumnos.
- La comunicación entre alumnos y grupos es escasa o nula..
- Los alumnos no están acostumbrados a compartir sus desarrollos o soluciones.
- El seguimiento de los subproductos de un desarrollo es complejo.
- No existen recursos software (ni hardware) que permitan el seguimiento y gestión de la gran cantidad de información y documentación que se genera en el desarrollo de un proyecto software.
- La inmensa cantidad de información que se genera es tediosa de manejar, compleja de analizar y difícil de evaluar por parte del profesor.

Se hace necesario para la eficiente implantación del modelo ECTS el dotar a los docentes de herramientas adecuadas que le permitan aplicar las directivas propuestas. El desarrollo de estas herramientas es largo y complejo, dado que las Universidades no invierten los recursos necesarios para ello, no existe la implicación necesaria de los servicios de informáticos con los que dispone la Universidad, y no existe una “conciencia” real de la necesidad de los mismos.

Por otra parte, la aplicación del modelo ECTS está suponiendo para el profesorado un gran esfuerzo, dedicando mucho tiempo a su formación y a la aplicación del nuevo modelo de enseñanza, lo que no se está viendo recompensado ni por los resultados obtenidos, ni por los medios que la Universidad pone a su disposición para ayudarle en esta tarea, ni por el reconocimiento dado a este esfuerzo.

Con este proyecto se ha pretendido estudiar y desarrollar, una herramienta software (aunque en un estadio de primera versión o versión beta) que facilite la labor docente del profesorado, que le aporte información veraz y completa del trabajo de los alumnos, y que le permita poner en práctica las nuevas directivas de enseñanza, incentivando diferentes habilidades y actitudes en los alumnos.

Si bien nuestro objetivo es que esta herramienta abarque la docencia práctica de cualquier asignatura, hemos centrado nuestra atención en los estudios de Informática. La docencia práctica consiste, de forma general, en la aplicación de los conceptos teóricos impartidos a casos prácticos (simulando casos reales) mediante el uso del ordenador. Esta tarea o actividad es realizada por los alumnos de forma individual o grupos pequeños, generando un producto que es evaluado por el profesor. Este producto puede ser documental (documentación formal de la solución), una aplicación o programa y, en la mayoría de los casos, ambos.

El profesor tiene cada curso académico, para cada una de las asignaturas asignadas, que:

1. Proponer una serie de problemas prácticos, generalmente diferentes a los de cursos previos.
2. Seguir el trabajo diario de los alumnos, resolver sus dudas, realizar correcciones, proponer soluciones, analizar los subproductos (documentales y software) que se van generando, etc.
3. Evaluar el trabajo realizado, lo que supone el estudio exhaustivo de los productos generados por los alumnos.

## 2. Objetivos

*El objetivo planteado en este proyecto es el análisis, diseño y desarrollo de una herramienta software mediante la cual se pueda llevar a cabo el desarrollo, seguimiento y control de las actividades prácticas de los alumnos universitarios, permitiendo incentivar la enseñanza de técnicas, herramientas y actitudes marcadas como objetivos en el modelo EEES.*

La herramienta estará abierta a su adaptación a cualquier tipo de enseñanza práctica universitaria, aunque en este proyecto se ha abordado bajo la problemática de las titulaciones de Informática, podrá estar integrada dentro del soporte que la universidad ofrece a través de Internet/Intranet, funciona de forma similar a un gestor de contenidos, hace uso de estándares y de software de dominio público, y será cedida al Vicerrectorado de Calidad para su estudio, evaluación y posterior completitud por la Universidad, si así lo estima oportuno.

Los objetivos, tareas y actividades docentes que abarca la herramienta principalmente son:

- Permite la creación de nuevas actividades docentes prácticas relacionadas con asignaturas (Proyectos).
- Permite la asignación de docentes y alumnos a cada proyecto.
- Permite la configuración de los proyectos, estableciendo fases, actividades, tareas, así como calendarios.
- Permite la asignación de recursos (alumnos) a las diferentes fases, actividades y tareas, y la asignación de responsabilidades a los mismos.
- Establece los mecanismos y medios de difusión de los trabajos y resultados.
- Establece los mecanismos y medios de corrección, evaluación, mejora de las diferentes actividades realizadas.
- Establece los estándares a utilizar para cada una de las tareas, las comunicaciones, correcciones, incidencias, etc.
- Permite el seguimiento de las diferentes actividades, su control y su difusión segura.
- Facilita el análisis y evaluación de las diferentes actividades y del trabajo realizado por alumnos en las mismas, generando informes que faciliten la tarea docente de evaluar no sólo en contenido y esfuerzo. sino también en actitudes.

## 3. Descripción de la experiencia

*El desarrollo del proyecto se ha llevado a cabo conforme al calendario prefijado, estando actualmente operativo un prototipo o versión beta resultado del trabajo realizado. Esto ha sido posible gracias a la colaboración de los alumnos en la realización de sus proyectos Fin de Carrera, en nuestro caso de la titulación de Ingeniería Informática. Sin ello, sin recursos de personal que sean sufragados con estos proyectos de Innovación y Mejora, consideramos no es posible el desarrollo de soluciones profesionales y eficientes y de impacto en la Innovación y Mejora Universitaria.*

Ya se ha mencionado la importancia de la existencia de herramientas cuya finalidad sea difundir actividades académicamente dirigidas dentro del marco de los créditos ECTS para la adaptación al Espacio Europeo de Educación Superior.

Para la titulación de informática este cambio ha supuesto un gran esfuerzo, tanto por parte de los profesores como por parte de los alumnos, al igual que en el resto de las titulaciones; pero sin embargo, también implica un reto y la posibilidad de poder llevar a cabo lo aprendido

desde un punto de vista menos conceptual y más práctico, algo que es muy útil y necesario en las asignaturas de la titulación de informática.

En este sentido, el desarrollo de una herramienta que permita el seguimiento de los trabajos prácticos de los alumnos de la titulación de informática supone una gran ayuda para los profesores a la hora de plantear las asignaturas y el desarrollo de los trabajos, pero también proporciona una experiencia más motivadora para los alumnos, el verse implicados en un proyecto más verosímil y con la facilidad para poder interactuar que proporciona una herramienta software.

Al mismo tiempo, la Gestión de la Configuración implica una garantía de calidad del software, por lo que estaríamos hablando de *una herramienta útil tanto desde el punto de vista funcional como metodológico*.

La herramienta desarrollada proporciona a los usuarios la funcionalidad necesaria para el desarrollo de sistemas software, su seguimiento y control, en un trabajo colaborativo y siguiendo unos estándares y calendario establecido.

Esta funcionalidad se proporciona al usuario a través de una interfaz gráfica sencilla e intuitiva. Esta interfaz está dividida en varias áreas/módulos con restricciones de acceso en función del tipo de usuario y privilegios.

La actividad de gestión de la configuración se lleva a cabo de manera semiautomática, de manera que la herramienta proporciona los recursos necesarios para poder llevar a cabo el seguimiento y control de desarrollo software, pero son los profesores los que utilizan de un modo personalizado estos recursos para adaptarlos a sus necesidades.

El usuario una vez que haya ingresado en la herramienta con los privilegios pertinentes debe tener acceso a una serie de menús e ítems suficientemente descriptivos y completos, a través de estos menús e ítems los alumnos accederán a la información facilitada por el profesor/profesores u otros alumnos, pudiendo ser esta información tanto de tipo textual como ficheros para descarga o incluso accesos a la base de datos en el que se almacenarán los diferentes proyectos con sus correspondientes fases y tareas e información de otros usuarios de la herramienta.

Los usuarios tienen limitadas sus acciones en la herramienta en función del tipo de usuario que sean con un nivel de privilegios dentro de la misma. De este modo, habrá usuarios que puedan acceder de forma completa, con total libertad para incluir artículos, importar y exportar ficheros y acceder a todas las secciones, mientras otros tienen limitado el acceso a alguna de estas secciones o incluso a todas.

La herramienta de desarrollo ha sido planteada de modo que resulte sencilla e intuitiva, de manera que se facilite el acceso rápido a las diferentes funcionalidades ofrecidas, sin excesivas dificultades. De este modo será posible su utilización por cualquier usuario (ya sea profesor o alumno) que disponga de conocimientos informáticos fundamentales; ya que se pretende que pueda ser integrada dentro del entorno de universidad para facilitar el trabajo de alumnos y profesores en la adaptación a la docencia práctica de informática en el EEES.

Desde su planteamiento inicial esta aplicación ha sido concebida con la idea de que pueda ser útil desde un punto de vista práctico y que efectivamente sea usada en prácticas reales de la titulación de informática dentro de la Universidad de Córdoba, así pues su vida esperada variará en función del uso que finalmente se haga de la misma; no obstante como para cualquier sistema software será recomendable incorporar nuevas actualizaciones y funcionalidades, con el objetivo de incrementar su rendimiento.

La calidad y la fiabilidad son dos conceptos que pueden utilizarse a la hora de proporcionar una medida relativa al éxito de la aplicación implementada. Ambos términos presentan una gran importancia a la hora de realizar el desarrollo de sistemas informáticos, puesto que es

necesario proporcionar al usuario final una garantía que le permita depositar su confianza en el producto, puesto que en caso contrario podrían surgir reacciones adversas a su utilización.

Las diferentes etapas en que se ha organizado el proceso de desarrollo de la herramienta objeto del proyecto son las que se describen a continuación.

- **Fase de Preparación:** a lo largo de esta fase se han realizado labores relacionadas con el estudio de los temas objeto de desarrollo del proyecto, como son: Estudio de las funcionalidades de los Sistemas de Gestión de Contenido; Estudio de las diferentes opciones de Sistemas de Gestión de Contenido existentes para la elección del más adecuado; Estudio de algunas herramientas de Gestión de la Configuración para comprender su utilidad y ámbitos de aplicación; Estudio y análisis de la problemática del seguimiento y gestión de los trabajos prácticos de desarrollo de software que se llevan a cabo en las titulaciones de informática. Una vez sentadas las bases necesarias, se ha procedido a integrarnos en el estudio del problema objeto del proyecto, conociendo las restricciones del mismo.

- **Fase de Diseño:** Esta fase encierra cuatro etapas, la primera es el Diseño de los Datos que transforma el modelo de dominio de la información, creado durante el análisis, en las estructuras de datos necesarios para implementar el software. La segunda etapa es el Diseño Arquitectónico que define la relación entre cada uno de los elementos estructurales del programa. El Diseño de la Interfaz, es la tercera etapa, en la cual se describe como se comunica el software consigo mismo, con los sistemas que operan junto a él y con los operadores y usuarios que lo emplean. El Diseño de Procedimientos, es la cuarta etapa de esta fase, y transforma elementos estructurales de la arquitectura del programa. El Diseño debe proporcionar una completa idea de lo que es el software, enfocando los dominios de datos, funcional y comportamiento desde el punto de vista de la Implementación. Por tanto, en esta fase se desarrollará una estructura de datos adecuada para la implementación del problema, así como también los métodos necesarios para la resolución de nuestro problema. Además es en esta fase donde debemos realizar el diseño de la interfaz de usuario.

- **Fase de Implementación:** Es esta tercera fase a la que corresponde la implementación de la aplicación, teniendo en cuenta las estructuras de datos y métodos diseñados en la fase de diseño, es decir, corresponde a esta fase la puesta en práctica de lo ya diseñado anteriormente. Llegados a este punto, tras la fase de diseño, la implementación se realiza construyendo un modelo de diseño en base al modelo de análisis anterior; siendo esta implementación del sistema un mapeo del modelo anterior a un modelo de más bajo nivel.

- **Fase de Documentación:** Las actividades de esta etapa se llevan a cabo durante todo el desarrollo del sistema, en ella debemos realizar toda la documentación relativa al proyecto.

Para el desarrollo de esta herramienta se ha contado con el respaldo de un sistema de gestión de contenido, aprovechando todas las ventajas que ello conlleva, siendo una de las más importantes la seguridad que éste aporta. Por un lado un sistema de gestión de contenido permite manejar de manera independiente el contenido y el diseño. Así, es posible manejar el contenido y darle en cualquier momento un diseño distinto al sitio sin tener que darle formato al contenido de nuevo, permitiendo la publicación controlada. Estos sistemas proporcionan diferentes niveles de acceso dependiendo del tipo de usuario lo que garantiza el resguardo de los datos.

Se puede añadir incluso un nuevo condicionante a esta metodología, que es la accesibilidad, un aspecto muy de moda últimamente en temas relacionados con los portales Web; y en este sentido se puede decir, que el apoyo de un sistema gestor de contenidos es fundamental, ya que aporta herramientas para que los creadores sin conocimientos técnicos en páginas Web puedan concentrarse en el contenido y puede proporcionar la comodidad de acceso desde cualquier ordenador con un navegador y posibilidad de acceder a Internet.

En cuanto a la escalabilidad, si se usa un sistema de gestión de contenido para el desarrollo de la herramienta se trabajará con la ventaja de que los gestores de contenido aportan herramientas para definir la estructura, el formato, el aspecto visual, uso de patrones, y un sistema modular que permite incluir funciones no previstas originalmente.

#### 4. Materiales y métodos

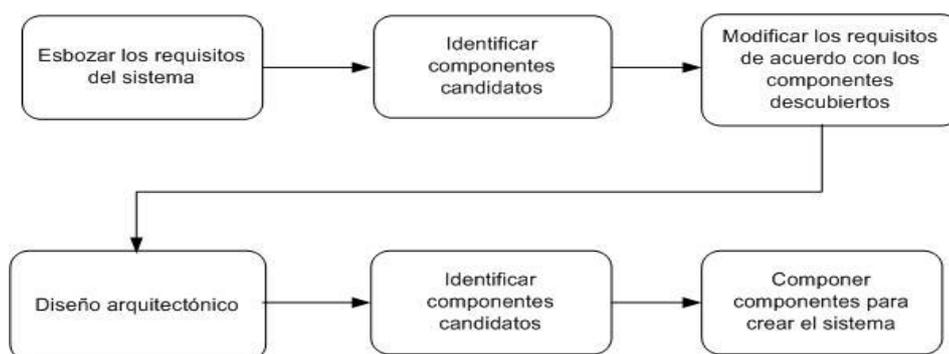
Para el desarrollo del trabajo se han utilizado los siguientes recursos:

- Ordenadores personales (PCs) con sistema operativo Windows XP existentes en el grupo de investigación ISCBD..
- Joomla, como sistema de gestión de contenidos de código abierto.
- MySQL, como sistema de gestión de bases de datos.
- PHP, como lenguaje de programación para el desarrollo y modificación de extensiones y componentes para la herramienta.

Debido a que se ha utilizado un sistema gestor de contenidos como apoyo para el desarrollo de la herramienta, el desarrollo software no se ha adaptado a ninguna metodología tradicional; sino que se ha seguido una metodología de desarrollo basada en componentes. Ya que en la actualidad el desarrollo de software basado en componentes se ha transformado en el componente más seguro tanto para la construcción de complejos sistemas como para aplicaciones de software.

El Desarrollo de Software Basado en Componentes (DSBC) trata de apoyar las bases para el diseño y desarrollo de aplicaciones distribuidas basadas en componentes software reutilizables. Esta norma se ha convertido en la actualidad en una metodología de gran utilidad, tanto desde el panorama académico como desde el industrial, en donde su demanda de temas es cada día mayor. Así pues, la finalidad primordial del DSBC es entender los diferentes modelos de desarrollo de software y la importancia de sus componentes y servicios.

La ingeniería del software basada en componentes (ISBC) es un proceso (ver figura 1) que se centra en el diseño y construcción de sistemas software que utilizan “componentes” reutilizables. La ISBC está cambiando la forma en que se desarrollan los sistemas, representa



**Figura 1.** Principales Actividades del Proceso de ISBC

la filosofía de “comprar, no construir”, que expusieron Fred Brooks y otros. De la misma manera que las primeras subrutinas liberaban al programador de tener que pensar en detalles, la ISBC cambia su objetivo y pasa de programar el software a componer sistemas software. La implementación ha dado paso a la integración como núcleo del enfoque. Se puede decir que en su base se encuentra la suposición de que en muchos sistemas grandes de software existe una base común suficiente como para justificar los componentes reutilizables para explotar y satisfacer a esa base común.

En el desarrollo de este proyecto se ha intentado evitar la fase de modificación de los requisitos en la medida de lo posible ya que, aunque Sommerville la considere una de las fases del proceso, hay que tener en cuenta que durante el desarrollo se trata de adaptar el sistema a los requisitos y no los requisitos al sistema.

Los requerimientos del usuario se desarrollan inicialmente en forma de esquema en lugar de con detalle, y los “stakeholders” son alentados a ser lo más flexibles posible en la definición de sus requerimientos. La razón de esto es que los requerimientos muy específicos limitan el número de componentes que podrían satisfacer estos requerimientos. Sin embargo, y a diferencia del desarrollo incremental, se necesita un conjunto completo de requerimientos con el fin de que se puedan identificar para su reutilización tantos componentes como sea posible.

Por lo tanto en el análisis de requisitos del sistema hemos determinado los siguientes dominios para el correcto funcionamiento de la herramienta: la estructura y apariencia de la misma, los elementos fundamentales a considerar, los tipos y privilegios de los usuarios, la gestión y administración de la información del sistema, los espacios de carga y descarga de información y la gestión y manipulación de las notificaciones de incidencias y cambios.

Una vez recogidos y validados todos los requisitos del sistema se modeló la funcionalidad de la herramienta mediante los Diagramas de Casos de Uso de UML considerando los distintos perfiles de usuario y la funcionalidad que se le permitirá en el sistema.

El siguiente paso en la metodología aplicada es determinar la arquitectura del sistema gestor de la base de datos y para ello se desarrollo el modelo conceptual que podemos observar en la figura 2 a partir del cual se ha obtenido el lógico y por último la estructura física de la arquitectura que soporta la herramienta.

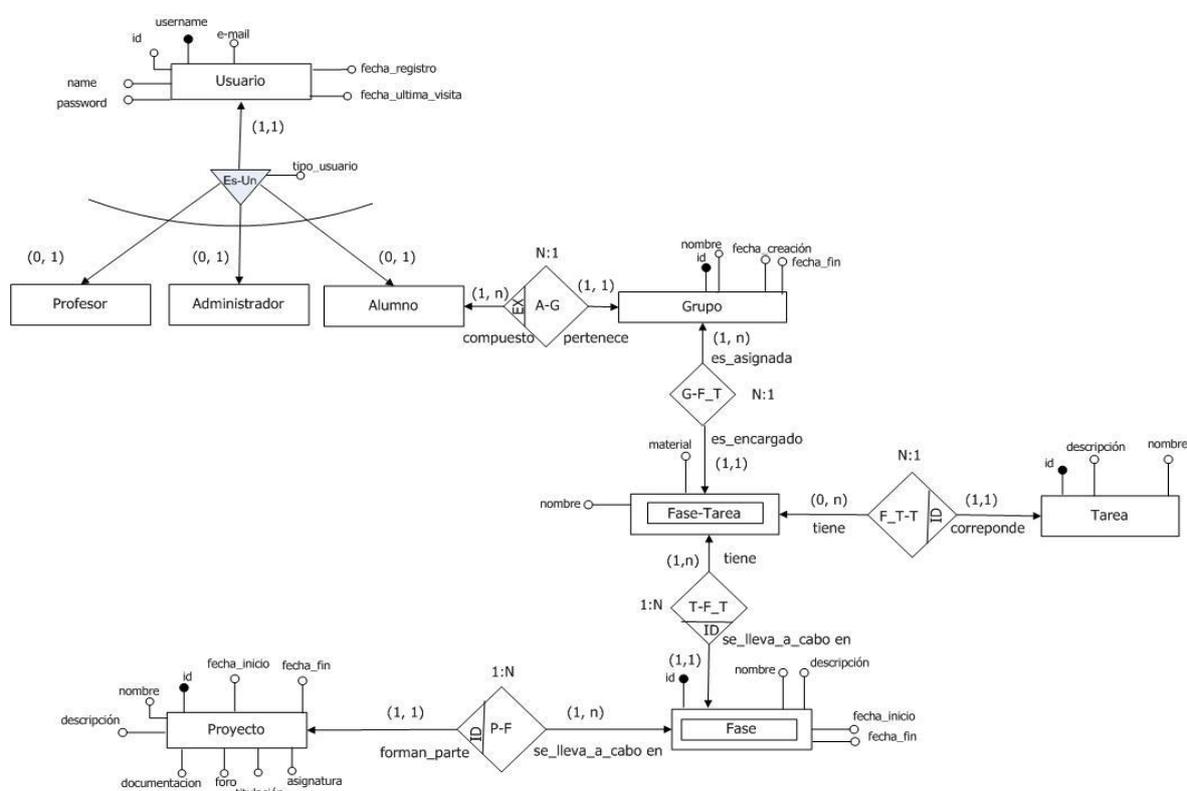


Figura 2. Modelo Conceptual del sistema (Modelo EER)

Con la estructura y necesidades necesarias para llevar a cabo los objetivos del desarrollo de la herramienta se ha realizado posteriormente siguiendo la metodología ISBC las actividades de búsqueda, selección validación y composición de componentes obteniendo

como resultado una versión inicial de la herramienta que constituía el objetivo y que se ha denominado GECO.

## 5. Resultados obtenidos y disponibilidad de uso

Se ha construido una herramienta en versión de evaluación que será utilizada por los alumnos de determinadas asignaturas de las titulaciones de informática tanto en su grado medio como superior que nos proporcionarán un periodo de pruebas en explotación si bien será algo limitado debido a que para no vulnerar la Ley Orgánica de Protección de Datos se utilizará siempre con el conocimiento y permiso de los alumnos implicados en la experiencia.

En la figura 3 se muestra la pantalla de presentación de la herramienta desarrollada como resultado del proyecto que cumple con los objetivos marcados al inicio del mismo respecto a su utilidad, facilidad de uso y criterios de calidad que se han corroborado mediante las pruebas pertinentes del sistema.



Figura 3. Pantalla de Inicio

A resaltar es que se ha proporcionado una internacionalización de la herramienta en cuanto a su contenido de información permitiendo su uso por parte de una comunidad de alumnos de mayor alcance.

Como conclusión se ha desarrollado una herramienta para la Gestión de Configuración del software y para el seguimiento de los trabajos prácticos de los alumnos de informática de la Universidad de Córdoba mediante un portal Web.

Cabe destacar que la herramienta desarrollada carece de determinada funcionalidad que le proporcionaría mayor utilidad y en perfil más ajustado a los objetivos de las prácticas en determinadas asignaturas de informática debido a diferentes razones como: no estar cubiertas por los objetivos del proyecto, falta de presupuesto o bien debe ser abordada a nivel corporativo.

## 6. Utilidad

La utilidad del producto generado en este proyecto es una herramienta que nos permitirá el desarrollo, seguimiento y control de las actividades prácticas de los alumnos universitarios y en este caso concreto de los alumnos de las Ingenierías Informáticas.

Por tanto mediante la herramienta desarrollada se permitirá:

1. La creación de los denominados “proyectos” para el desarrollo de las actividades docentes prácticas de las diferentes asignaturas. Proporcionando una asignación jerarquizada o por niveles según el tipo de usuario (profesor, alumno o administrador).
2. La configuración de los proyectos mediante el establecimiento de fases, actividades y tareas dentro de cada uno de ellos, así como la asignación de recursos, calendario y de responsabilidades.
3. El establecimiento de mecanismos para la corrección, evaluación y mejora de las diferentes actividades realizadas.
4. Posibilitar el verificar el cumplimiento de los estándares establecidos para las diferentes actividades.
5. El conocimiento y puesta en práctica por parte del alumnado de protocolos muy parecidos a los que encontrará en un futuro en el mercado de trabajo.

## **7. Observaciones y comentarios**

Los proyectos de mejora permiten el desarrollo a nivel de prototipado, pero debido a las limitaciones en su financiación no puede llegar a desarrollarse proyectos profesionales y con una calidad y funcionalidad más o menos compleja. Esta situación se produce debido a los siguientes inconvenientes:

- Debido a la limitación económica no es posible la contratación de personal de apoyo que realice labores que a los docentes les es prácticamente imposible desempeñar debido a las limitaciones de tiempo y que por otra parte al ser más rutinarias y técnicas pueden ser dirigidas por el docente y realizadas por dicho personal.
- La gestión con respecto a los gastos ocasionados a cargo de la dotación del proyecto es muy restrictiva lo que ocasiona que no se pueda realizar las compras en las empresas más competitivas respecto al precio.

Para subsanar estos inconvenientes consideramos que sería necesario replantearse la convocatoria de proyectos intentando adaptarlas más a las necesidades reales y que los resultados obtenidos de los proyectos puedan ser útiles a la comunidad universitaria.

## **8. Autoevaluación de la experiencia**

La evaluación de la experiencia por parte del equipo participante ha sido favorable en cuanto a los resultados obtenidos teniendo en cuenta las condiciones limitadas en cuanto a tiempo y financiación. Sin embargo la utilidad del trabajo realizado se observará ampliamente en el curso académico 08/09 en el cual utilizaremos la herramienta para llevar a cabo los procedimientos y prácticas, ya utilizados en las asignaturas, de manera automatizada.

## **9. Bibliografía**

1. Babich, W. Software Configuration Management. Addison-Wesley, 1986.
2. Berlack, H. Ronald. Software Configuration Management (Wiley Series in Software Engineering Practice). John Wiley & Sons, 1991.
3. Ministerio de Educación y Ciencia. Espacio Europeo de Educación Superior. <<http://www.mec.es/universidades/ees/index.html>>. [Consulta: Enero 2008]
4. Content Management Systems. <<http://www.cmsinfo.org>> [Consulta: Enero 2008]
5. Joomla! Spanish. <<http://www.joomlaspanish.org/>> [Consulta: Enero 2008]

6. Formación E-learning. <<http://www.formacionelearning.com/>> [Consulta: Enero 2008]
7. Component Software: Beyond Object- Oriented Programming, 2ª edición. Addison Wesley, 2002.
8. Shneiderman, B. Designing the User Interface, 3rd Edition. Addison-Wesley, 1998 en Sommerville, I. *Ingeniería del Software*, 7ª edición. Addison-Wesley, 2005.

Córdoba a 11 de Septiembre del 2008

Fdo: Irene Luque Ruiz