

UNIVERSIDAD AUTONOMA DE MADRID

ESCUELA POLITECNICA SUPERIOR



Grado en Ingeniería Informática

TRABAJO FIN DE GRADO

VIRTUALIZACIÓN DE GUÍAS DOCENTES

NOMBRE: Alejandro García Blanco

TUTOR: Fernando Díez Rubio

ENERO 2016

VIRTUALIZACIÓN DE GUÍAS DOCENTES

AUTOR: Alejandro García Blanco

TUTOR: Fernando Díez Rubio

**Dpto. de Ingeniería Informática
Escuela Politécnica Superior
Universidad Autónoma de Madrid
ENERO 2016**

Resumen

El mundo relacionado con las tecnologías está en constante evolución, apreciándose día a día los avances que se introducen en nuestras vidas a través de los teléfonos móviles, las televisiones inteligentes, los ordenadores, las tabletas, etc. Esta evolución se ve reflejada en multitud de aspectos y servicios que en muchas ocasiones nos pasan desapercibidos, pero que están presentes en la realidad que nos rodea.

Casi para la totalidad de problemas de todo tipo que surgen hoy en día, organizativos, de comunicaciones, técnicos, sanitarios, etc. existe una solución mediante un programa para un dispositivo móvil, una aplicación web para manejar contenidos en Internet o un nuevo sistema que permite realizar numerosas funciones en nuestra televisión. De ahí que las empresas hayan mostrado un gran interés en el sector de las TIC donde la demanda de productos y servicios no para de crecer.

En este contexto de avance tecnológico el objetivo principal que se plantea en este proyecto es el de ofrecer una solución al problema que supone actualmente la administración de las guías docentes de una titulación universitaria en el nuevo Espacio Europeo de Educación Superior (EEES). La administración se debe realizar entre varias personas que deben coordinarse entre sí, realizar modificaciones consensuadas sobre los documentos y aprobar los cambios con anterioridad a cada nuevo curso académico. El proyecto en cuestión pretende ofrecer una aplicación web de manejo sencillo, intuitivo, y que permita ahorrar tiempo.

Palabras clave

Guía Docente, Aplicación Web, Grado, Internet, Tecnología.

Abstract

The world related to technology is in constant evolution, and nowadays it can be seen in how our lifestyle has progressed through mobile phones, smart televisions, computers, tablets, etc. This evolution is reflected in the multitude of key aspects and services that, in many occasions, go by unnoticed, but are present in everything that surrounds us.

For most of the different problems that arise today, organizational, communication, technical, sanitary, etc. there exists a solution through an application for a mobile device, a web application to manage contents on the Internet, or a new system that allows us to carry out numerous functions on our television. It comes as no surprise that companies have shown interest in the computer technology area where the demand for products and services is in increase.

In this context of technologic progress the main objective set out in this project is to offer a solution to the problem that comes with the present management of the syllabus of a college degree in the new EEES. This management has to be carried out by several individuals. They coordinate among themselves to make modifications after arriving at an understanding about the documents. They also approve these changes after the new academic course. This project offers an alternative through a web application of easy and intuitive use to save time.

Keywords

Syllabus, Web Application, Degree, Internet, Technology.

INDICE DE CONTENIDOS

1 INTRODUCCIÓN	1
1.1 OBJETIVOS.....	3
1.2 FASES DEL PROYECTO	5
1.3 ASIGNATURAS RELACIONADAS	7
2 CONTEXTO DEL PROYECTO	9
2.1 APLICACIONES WEB.....	9
2.2 HERRAMIENTAS UTILIZADAS	11
3 ANÁLISIS DE REQUISITOS	13
3.1 REQUISITOS FUNCIONALES	13
3.1.1 <i>Gestión de usuarios</i>	13
3.1.2 <i>Gestión de asignaturas/titulaciones</i>	14
3.1.3 <i>Gestión de bloques de una guía docente</i>	15
3.1.4 <i>Gestión de mensajes</i>	15
3.1.5 <i>Gestión de guías docentes</i>	16
3.2 REQUISITOS NO FUNCIONALES	16
4 DISEÑO DE LA SOLUCIÓN	19
4.1 DISEÑO DE LA INTERFAZ	19
4.2 DISEÑO DE LA ESTRUCTURA DE DATOS	21
4.3 CASOS DE USO	24
5 IMPLEMENTACIÓN	37
5.1 INTERFAZ DE USUARIO.....	37
5.2 BASE DE DATOS	39
5.3 FUNCIONALIDAD.....	41
6 PRUEBAS Y RESULTADOS	44
6.1 PRUEBAS UNITARIAS	45
6.2 PRUEBAS DE INTEGRACIÓN	47
6.3 PRUEBAS DE SISTEMA	49
6.4 PRUEBAS DE VALIDACIÓN.....	49
6.5 OTRAS PRUEBAS.....	50
6.6 RESULTADOS	51
7 CONCLUSIONES Y TRABAJO FUTURO	52
7.1 TRABAJO FUTURO	53
BIBLIOGRAFÍA	54
GLOSARIO	56

INDICE DE FIGURAS

ILUSTRACIÓN 1: GRADO EN INGENIERÍA INFORMÁTICA EN LA UNED	3
ILUSTRACIÓN 2: MODELO DE CICLO DE VIDA EN CASCADA	6
ILUSTRACIÓN 3: DEFINICIÓN DE UNA APLICACIÓN WEB.....	9
ILUSTRACIÓN 4: DIAGRAMA DE FLUJO DE NAVEGACIÓN.....	20
ILUSTRACIÓN 5: DISEÑO DE LA BASE DE DATOS.....	23
ILUSTRACIÓN 6: DIAGRAMA DE CASOS DE USO	25
ILUSTRACIÓN 7: INTERFAZ DE INICIO	37
ILUSTRACIÓN 8: EDITOR NICEDIT.....	39
ILUSTRACIÓN 9: APLICACIÓN PHPMYADMIN	40
ILUSTRACIÓN 10: VISUALIZACIÓN DE LA GUÍA DOCENTE.....	43
ILUSTRACIÓN 11: TIPOS DE PRUEBAS REALIZADAS EN EL PROYECTO	45

1 Introducción

En el mundo en que vivimos, los ordenadores van introduciéndose más en nuestro día a día, con el fin de facilitar las diferentes tareas que se nos presentan. Desde una sencilla aplicación que pueda administrar las mesas de un bar, hasta una base de datos que maneje la ingente cantidad de datos que pueda tener una gran empresa, se ve reflejada la necesidad de incorporar la evolución que está teniendo la tecnología para poder realizar avances en los distintos aspectos de nuestras vidas.

Más concretamente, en un ámbito docente, han ido surgiendo diferentes necesidades a la hora de evaluar, compartir recursos, o establecer una comunicación entre profesor y estudiante. Esto puede ser el caso de *Moodle*, la plataforma utilizada en la UAM (Universidad Autónoma de Madrid) para reunir todas las asignaturas de cada una de las titulaciones que ofrece.

Esta plataforma tiene diferentes usos, pero entre los más comunes en la EPS (Escuela Politécnica Superior) se encuentran la subida de diferentes recursos por parte del profesorado para impartir la materia, entrega de ejercicios y prácticas, y *feedback* con noticias sobre novedades en la asignatura, como puede ser el resultado de una evaluación, tanto de un examen como de una práctica.

En dicha plataforma, dentro de cada asignatura, se encuentra un documento que recoge todo el contenido de la materia, así como los métodos de evaluación y requisitos para sacar adelante dicha asignatura. Este documento es denominado la **Guía Docente** de la asignatura, y no es un elemento de la plataforma *Moodle* sino de las titulaciones y se publicita en las páginas web de los centros.

La guía docente es el instrumento que, desde el inicio de la nueva estructura de titulaciones adoptada a partir de la entrada en vigor del EEES, sirve de documento de planificación de toda la actividad docente, así como documento

informativo, que contiene información del equipo docente, métodos de evaluación, bibliografía, etc. Dicho documento establece un contrato entre el estudiante y el profesorado que es redactado antes del comienzo del curso académico, y no puede ser modificado durante el transcurso del mismo, a menos que la Junta del Centro lo apruebe. En la realización del mismo participa todo el equipo docente que compone una asignatura. Es por ello que necesitan coordinarse y compartir opiniones, siendo normalmente el método de comunicación el correo electrónico. Esto hace que sea una tarea lenta e incluso demandante para los profesores que tienen que estar pendientes de las guías docentes de varias asignaturas, así como los coordinadores de las mismas.

El objetivo principal del proyecto que presentamos ha consistido en desarrollar una aplicación que facilite el proceso de creación, edición y gestión de las guías docentes de una universidad. Se pueden encontrar distintas aplicaciones de cometido parecido, como puede ser en la UM (Universidad de Murcia), UPV (Universidad Politécnica de Valencia), o UNED (Universidad Nacional de Educación a Distancia, donde se puede visualizar las guías docentes de las distintas asignaturas de las titulaciones que imparten, y hacerse con una copia en formato descargable. Pero con esta aplicación se pretende dar un paso más, permitiendo reunir todas herramientas necesarias para poder llevar a cabo las diferentes tareas que implica la administración de las guías docentes. Para ello se dará uso a la estructuración de las guías docentes por bloques, que definen los aspectos a tener en cuenta en una guía docente, como son introducción, datos del equipo docente, objetivos del curso, contenidos del programa, bibliografía, métodos docentes, cronograma, entre otros.

The screenshot shows the UNED website interface. At the top, there is a navigation bar with links for 'Estudios', 'Investigación', 'Internacional', 'Comunicación', 'La UNED', and 'UNED Abierta'. On the right, there are links for 'Español | English' and 'Contacta'. Below the navigation bar, there is a search bar and a 'Acceso al CAMPUS' button. The main content area is titled 'Guía del Grado' and 'GRADO EN INGENIERÍA INFORMÁTICA'. Under 'Guía del Grado', there is a list of links for 'Curso 2014/2015', including 'Presentación', 'Competencias', 'Perfil Ingreso', 'Curso Adaptación', 'Plan de estudios', 'Estructura', 'Listado de Asignaturas', 'Prácticas', 'Calendario de implantación y extinción del plan antiguo', 'Reconocimientos de créditos', 'Calendario de exámenes', 'Normativa', 'Perfil de egreso', 'Salidas Profesionales', 'Documentación Oficial del Título', 'Sistema Interno de Garantía de Calidad del Título', 'Proyecto Fin de Grado', 'Buzón de Sugerencias, Reclamaciones y Felicitaciones', and 'Estudiantes con discapacidad'. The 'PRESENTACIÓN' section contains text about the degree's objectives and the university's commitment to research and innovation. The 'Estudia en el EEES en la UNED' section mentions the university's accreditation.

Ilustración 1: Grado en Ingeniería Informática en la UNED

1.1 Objetivos

A lo largo de cualquier carrera, el estudiante afronta diferentes retos, que se presentan en forma de exámenes, prácticas, o pequeños proyectos. Estas tareas le servirán para formarse, aprender a afrontar los distintos problemas que se le pueden aparecer, así como proporcionarle una nueva perspectiva de ver el mundo en el vivimos.

Gracias a la formación recibida y a las distintas tareas realizadas durante todo el grado, se está en disposición de acometer proyectos de envergadura profesional, en la que la extensión, la dificultad o la combinación de ambas hagan que el trabajo a realizar tenga una entidad o dificultad importantes. En este sentido, el proyecto que se desarrolla en el presente trabajo de fin de grado tiene por objetivo abordar el problema actual que se plantea a la hora de

crear y mantener las guías docentes de las asignaturas de cualquier titulación. Un trabajo de estas características conlleva diferentes tareas de análisis, diseño, planificación y desarrollo, buscando, como fin último, proporcionar a los usuarios una aplicación que facilite la gestión de las Guías Docentes.

El TFG (Trabajo de Fin de Grado) representa para el estudiante la prueba definitiva, dónde se aplica todo lo aprendido a la hora de participar en el desarrollo de las distintas etapas de un proyecto. Este TFG pretende ofrecer una visión de lo que una aplicación web puede aportar cuando se busca solucionar un problema que puede parecer sencillo a primera vista. El problema reside, en particular, en la dificultad actual que tienen los equipos docentes de cualquier asignatura de cualquier titulación de las que se imparten en las facultades y en la Escuela Politécnica de la UAM para poder realizar una guía docente.

De acuerdo a este problema, el objetivo general del trabajo ha consistido en la creación de una aplicación web mediante la cual los profesores puedan registrarse y modificar, de un modo sencillo y transparente a todo el equipo, las guías docentes que les corresponden. Estos cambios se verán reflejados para los demás profesores que pertenezcan al equipo docente de la asignatura, notificándoles del cambio realizado.

Como objetivos específicos, entre las distintas funcionalidades que integra la aplicación web desarrollada hemos de mencionar:

- Sistema de registro y conexión del usuario a la aplicación, dando de alta a los distintos componentes del equipo docente, registrando las asignaturas y las titulaciones, entre otras.
- Buscador de asignaturas y titulaciones
- Editor de las guías docentes específicas de cada asignatura.
- Validación, por un usuario con los privilegios necesarios en la aplicación, que permitirá asignar un código identificador único impidiendo así la modificación y falsificación de las mismas.
- Impresión en formato PDF de las Guías Docentes.

Los usuarios se diferenciarán a partir de la especificación de unos privilegios, siendo los coordinadores los que pueden modificar las guías docentes de una determinada asignatura, los profesores los que tienen acceso a las guías docentes y los cambios pero no pueden realizarlos ellos mismos, y los invitados que serán aquellos que pueden acceder a la aplicación sin acceder al sistema de usuarios y que podrán visualizar las guías docentes y obtenerlas en formato descargable.

1.2 Fases del proyecto

Durante el desarrollo del proyecto se han seguido diferentes etapas para optimizar el proceso y el producto software desarrollado. Dichas etapas coinciden a grandes rasgos con las que se realizan en la mayoría de proyectos software:

- **Análisis:** en esta fase se establecen los requisitos, que nos permitirán tener una perspectiva de cómo va a funcionar la aplicación. Se valora la prioridad e importancia de cada requisito para poder garantizar su correcto funcionamiento antes de pasar a las fases de diseño y desarrollo.
- **Diseño:** tras la fase de análisis, se necesita realizar el diseño completo de la aplicación. En él se define qué estructura va a tener la base de datos para manejar todos los datos de la aplicación, así como el diseño de la interfaz de usuario para que este pueda navegar por la aplicación.
- **Implementación:** en esta fase se especifica la codificación de los diseños anteriormente definidos. Con una buena fase de análisis y diseño, la fase de implementación pasa a ser un trámite en el cual se escribe el código que permitirá el funcionamiento de la aplicación.
- **Pruebas:** en la fase de implementación se procede a realizar pruebas de funcionamiento, pero es en la fase de pruebas donde se tiene que completar el proceso para asegurarse que todas las funciones y módulos de la aplicación realizan correctamente su funcionamiento. Para ello se abarcan todos los posibles casos de uso y se documentan

los resultados obtenidos. Esto permitirá evitar situaciones en las que la aplicación tiene un rendimiento incorrecto que pueda afectar a la experiencia del usuario.

- **Documentación:** en esta fase se incluye toda la documentación de la aplicación, desde el análisis hasta las pruebas, recogiendo los aspectos más importantes del proyecto y las relaciones de los puntos claves de cada fase. Además incluirá las conclusiones y trabajo futuro que se podrá llevar a cabo en el proyecto.
- **Mantenimiento:** por último, pero no menos importante, en la fase de mantenimiento se procede a realizar mejoras, actualizaciones y modificaciones a la aplicación. Suele ser la fase más larga de un proyecto software y debido a que puede ser continuado por otra persona diferente a la que realiza el proyecto, la documentación juega un papel importante en esta fase.

Con las fases del proyecto definidas y los criterios de transición definidos entre cada una de ellas, el ciclo de vida utilizado es uno en cascada.

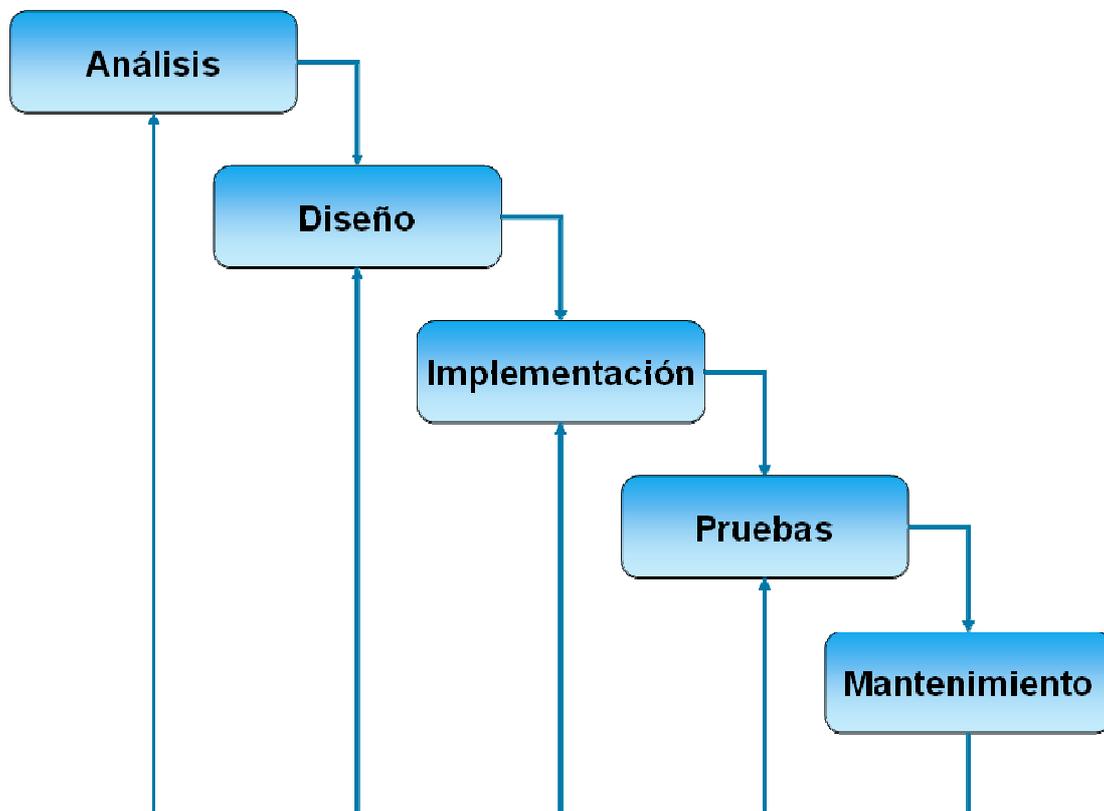


Ilustración 2: Modelo de ciclo de vida en cascada

Esto permite partir de la primera fase e ir añadiendo aspectos según se avanza en las distintas fases, ya que a lo largo de la realización del proyecto se han ido puliendo los distintos requisitos funcionales. Cada vez que se definen unos nuevos requisitos se procede a pasar por todas las etapas para poder adquirir el resultado esperado. Aunque a la larga puede consumir mucho tiempo, esto ha permitido un desarrollo continuo de la aplicación sin parones, aumentando así la productividad.

1.3 Asignaturas relacionadas

Durante el Grado de Ingeniería Informática se han cursado diferentes asignaturas que imparten unos conocimientos, los cuales algunos se pueden ver reflejados en este TFG. Por ello se procederá a realizar una numeración haciendo mención a dichas asignaturas y qué conceptos abarcan:

- **Estructuras de Datos:** se ha utilizado una base de datos y por tanto el lenguaje SQL para la realización de consultas a dicha base de datos, el cual se imparte en dicha asignatura.
- **Análisis y Diseño de Software / Proyecto de Análisis y Diseño de Software:** se han realizado durante el proyecto unas fases de análisis de requisitos y de diseño para poder llegar al resultado final de la aplicación, conceptos que se imparten en estas asignaturas.
- **Sistemas Informáticos I:** se trata de una aplicación web realizada en su totalidad en PHP, lenguaje con el cual se trabaja en las prácticas de dicha asignatura.
- **Ingeniería del Software / Proyecto de Ingeniería del Software:** todo producto software tiene un ciclo de vida. Las distintas fases de creación, producción y mantenimiento, así como documentación son competencias de estas asignaturas.
- **Redes de Comunicaciones II:** se ha utilizado para la validación de las guías docentes en formato descargable, el algoritmo de encriptación *sha1*, mencionado en el apartado de seguridad de esta asignatura.

El documento se ha dividido de forma que se expongan los puntos claves, características y decisiones importantes que respectan a cada fase:

- **Capítulo 2:** se habla del contexto del proyecto, dónde se proporciona al lector una visión general de la aplicación, así como las distintas herramientas utilizadas en ella.
- **Capítulo 3:** se procede a describir los requisitos tanto funcionales como no funcionales, y el análisis que se ha llevado a cabo para llegar a ellos.
- **Capítulo 4:** se expone el diseño de la solución teniendo en cuenta el análisis de requisitos realizado anteriormente además de describir los distintos casos de uso de la aplicación.
- **Capítulo 5:** se habla de la implementación llevada a cabo en la aplicación, llegando a describir la funcionalidad de la misma según el tipo de usuario.
- **Capítulo 6:** se comentan las pruebas llevadas a cabo para comprobar el funcionamiento de la aplicación.
- **Capítulo 7:** contiene las conclusiones sobre el proyecto así como un resumen subjetivo del desarrollo del mismo. Se habla también del trabajo futuro que puede realizarse en la aplicación teniendo en cuenta distintos aspectos de la misma.

2 Contexto del proyecto

Para la realización de este proyecto se necesita tener un mínimo de conocimientos previos respecto a las aplicaciones web, así como llevar a cabo una búsqueda de aplicaciones similares para poder obtener una idea genérica de lo que se pretende hacer.

Hoy en día las aplicaciones web están muy extendidas en el mundo laboral, especialmente en las empresas proveedoras de software que ofrecen sus productos vía web. Esto es debido a que el funcionamiento de las aplicaciones web debe ser independiente del entorno en el cual se ejecutan. Por lo que se evita crear distintos clientes para cada sistema operativo disponible o navegador web.

2.1 Aplicaciones web

Una aplicación web viene definida como una herramienta que el usuario puede utilizar accediendo desde Internet a través de un navegador. Dicho de otra manera se trata de una aplicación software que soportan los navegadores de Internet, lo que hace que sean populares al tratarse de un cliente ligero.



Ilustración 3: Definición de una Aplicación Web

Las empresas proveedoras de software han pasado a usar una estrategia en la que proveen acceso vía web al software que ofrecen. Esta estrategia se denomina Software como servicio, y las compañías que lo desarrollan, Proveedores de Aplicaciones de Servicio.

Al tratarse de aplicaciones web, el usuario no necesita descargar ni instalar ningún programa desde su ordenador, sino simplemente acceder a él a través de su navegador. Esto permite ahorrar tiempo y recursos, ya que no ocupa espacio en el sistema del usuario, además de evitar problemas de compatibilidad, necesitando simplemente un navegador actualizado. Además hay más ventajas de las que se benefician las empresas que utilizan esta estrategia:

- **Portable:** al accederse a través de un navegador, la aplicación web puede ser accedida desde cualquier sistema, incluso dispositivos móviles, lo cual amplía el mercado al cual puede estar destinado.
- **Disponible:** dependiendo del tamaño de la aplicación web, puede estar alojada en distintas localizaciones, asegurando la continuidad del mismo.
- **Seguridad:** como no se tiene que descargar una aplicación al sistema del usuario, el riesgo de que se infecte a través de la aplicación web es nulo como tal.
- **Actualizado:** el software es gestionado por el propio desarrollador, por lo que siempre se tendrá a disposición del usuario la última versión.
- **Futuro:** la mejora continua en los navegadores auguran mejoras de funcionalidad que facilitarán cada vez más la aparición de aplicaciones web más complejas.

Debido a todo lo mencionada anteriormente, las aplicaciones web pasan a ser una opción viable para las empresas. Además ofrecen una gran versatilidad a la hora de programarlas, ya que se pueden desarrollar en PHP, Java, Javascript, Perl, Ruby, etc.

2.2 Herramientas utilizadas

A lo largo del proyecto se han utilizado numerosas herramientas que han facilitado el desarrollo del mismo. También se han introducido herramientas que han permitido implementar funcionalidades específicas. A continuación todas las herramientas utilizadas en el desarrollo del proyecto:

- **XAMPP**: se trata de un servidor independiente de plataforma, software libre que consiste en el sistema de gestión de base de datos. Una vez configurado correctamente se procede a utilizarlo para hacer uso el servidor web Apache y la base de datos MySQL.
- **Apache**: es un servidor web HTTP de código abierto que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Se utiliza para establecer el entorno de pruebas local de la aplicación web.
- **MySQL**: sistema de administración de base de datos relacional, multihilo y multiusuario. Implementa el lenguaje SQL e interactúa con los lenguajes de programación más utilizados, como son PHP, Perl y Java. Se utiliza como motor de la base de datos de la aplicación.
- **Notepad++**: es un editor de texto y de código fuente libre con soporte para varios lenguajes de programación. Amplia la funcionalidad del Bloc de notas, incluyendo opciones que son muy útiles a la hora de programar como es la opción de sobresaltar distintos elementos de un lenguaje específico de programación.
- **NicEdit**: se trata de un editor ligero de plataforma que permite editar texto en una página web de manera sencilla y añadiendo funcionalidades de modificación de texto. Se utiliza como el editor para los textos de las guías docentes.
- **DomPDF**: es un conversor de HTML a PDF. Esto nos permite a partir del contenido de una página web obtener un fichero descargable. Se utiliza para las guías docentes.
- **phpMyAdmin**: es una herramienta de software libre escrito en PHP que mejora la administración de las bases de datos MySQL a través de una página web. Es compatible con la mayoría de operaciones que se pueden realizar en MySQL, haciendo de ella una herramienta

indispensable para el tratamiento de datos, y visualización de los mismos.

Además el proyecto se ha realizado utilizando dos lenguajes principalmente:

- **PHP:** se trata de un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. Se considera uno de los lenguajes más flexibles, potentes y de alto rendimiento.
- **SQL:** se trata de un lenguaje declarativo de acceso a base de datos relacionales que permite especificar diversos tipos de operaciones en ellas. Una de sus principales características el manejo del álgebra y cálculo relacional que permiten realizar consultas de forma sencilla.

3 Análisis de requisitos

Se denomina requisito a aquella circunstancia que se tiene que cumplir para poder especificar un comportamiento o funcionalidad de una aplicación. Este término es muy utilizado en las ingenierías en un sentido formal para definir lo que un sistema tiene que hacer. Para ello precisa ser necesario, sin ambigüedades, conciso, consistente y verificable ya que si no cumple ninguna de estas características no está bien formulado.

Se distinguen dos tipos principales de requisitos:

- **Requisitos funcionales:** este tipo de requisito define una función propia del sistema al que pertenece. Esta función expresa una capacidad de acción que tiene dicho sistema, es decir, una funcionalidad.
- **Requisitos no funcionales:** este tipo de requisito especifica un criterio que el sistema debe tener, imponiendo restricciones en el diseño o la implementación para cumplir con un cierto estándar de calidad.

3.1 Requisitos funcionales

En esta sección se definen los distintos requisitos funcionales, agrupados en subcategorías que permitirá la mayor comprensión de los mismos.

3.1.1 Gestión de usuarios

Para mayor transparencia y entendimiento de los usuarios, se establecen unos perfiles para diferenciarlos:

- **Coordinador:** usuario registrado que tiene privilegios adicionales en una asignatura concreta. Entre estos privilegios se encuentra la posibilidad de crear o modificar un bloque, o validar una guía docente en formato descargable.
- **Profesor:** usuario registrado que se encuentra ligado a una o varias asignaturas. Al igual que el coordinador, tiene acceso a los

mensajes que le permite estar notificado de los cambios que se realizan en una guía docente de una asignatura.

- **Invitado:** usuario que no necesita registrarse y por lo tanto acceder a la aplicación. A pesar de ello puede acceder a las asignaturas, visualizar las guías docentes e incluso obtenerlas en formato descargable.

En la gestión de usuarios se aborda todo lo referido a un usuario, desde su registro en la base de datos, a su acceso a la aplicación.

- **GU01:** Creación de un usuario: esto será posible a través del registro. Para registrar un nuevo usuario tendremos que acceder a la opción marcada como tal, e introducir la información necesaria (nombre de usuario, clave, dirección de correo electrónico, nombre completo y DNI). Dicha información se guardará en la base de datos para su posterior utilización.
- **GU02:** Conectarse a la aplicación como usuario registrado: comúnmente conocido como *Log-In*, el usuario introduce el nombre de usuario y clave que utilizó para registrarse. Una vez comprobados los datos el usuario se encontrará conectado a la aplicación pudiendo realizar acciones adicionales dependiendo de los privilegios que tenga.
- **GU03:** Desconectarse de la aplicación: comúnmente conocido como *Log-Out*, el usuario registrado tiene que haberse conectado a la aplicación anteriormente para poder salir de ella y volverse a encontrar como invitado.

3.1.2 Gestión de asignaturas/titulaciones

En la gestión de asignaturas y titulaciones se tiene la carga de datos donde se incluyen todas las asignaturas y titulaciones de la aplicación, además de la búsqueda de las mismas.

- **GA01:** Actualizar las asignaturas y titulaciones de la aplicación: se le permite al administrador, a través de un fichero de configuración,

cambiar las distintas titulaciones y asignaturas que ofrecerá la aplicación.

- **GA02:** Búsqueda de asignaturas y titulaciones: tanto para un usuario registrado como para un "Invitado" se le permitirá buscar entre las distintas titulaciones y asignaturas que ofrece la aplicación.

3.1.3 Gestión de bloques de una guía docente

En la gestión de bloques de una guía docente se tiene todo lo referido a la administración de los componentes de una guía docente, desde su creación hasta su visualización, pasando por la modificación de los mismos.

- **GB01:** Creación de un nuevo bloque: dentro de cada guía docente se puede crear un nuevo bloque siempre y cuando el usuario registrado tenga los privilegios para ello. Esto permite añadir nuevo contenido con un título identificativo.
- **GB02:** Modificación de bloques: dentro de cada guía docente se puede modificar un bloque ya creado anteriormente siempre y cuando el usuario registrado tenga los privilegios para ello.
- **GB03:** Visualización de bloques: tanto si es un usuario registrado o un "Invitado", se podrán visualizar los distintos bloques de una determinada guía docente.

3.1.4 Gestión de mensajes

En la gestión de mensajes se hace referencia a los mensajes que se utilizan dentro de la aplicación para notificar a los usuarios de un cambio en las guías docentes.

- **GM01:** Creación de mensajes: cada vez que un usuario cree o modifique un bloque en una guía docente de una asignatura, la aplicación distribuirá un mensaje automático con información del evento realizado a todos los usuarios asociados a la misma.

- **GM02:** Lectura de mensajes: el usuario registrado podrá acceder a los mensajes, si existen, y proceder a leerlos, marcándolos como "Leídos" para futuras referencias.

3.1.5 Gestión de guías docentes

En la gestión de guías docentes se puede obtener una copia de una guía docente, o validarla si se tienen los permisos para ello.

- **GG01:** Descarga de las guías docentes: a través de la aplicación se podrán descargar todas las guías docentes que se requieran en formato PDF.
- **GG02:** Validación de las guías docentes: los usuarios con permisos de coordinación podrán validar una determinada guía docente, creando así un código único que permitirá identificar a posteriori si la guía docente obtenida en formato descargable ha sido modificada de alguna manera y evitando la falsificación de la misma.

3.2 Requisitos no funcionales

Se incluyen los siguientes requisitos que definen al sistema, aplicando unos estándares de calidad:

- **RNF01:** Mantenibilidad: el diseño de la aplicación estará dividido en módulos que facilitarán su comprensión y la posibilidad de actualizaciones futuras. Además el código estará debidamente comentado y las pruebas serán claras y concisas para mostrar correctamente el funcionamiento de la aplicación.
- **RNF02:** Portabilidad: la aplicación se pretende que se pueda visualizar en cualquier navegador web siguiendo los estándares de PHP. Se necesitará una conexión a Internet que permita hacer uso de la aplicación.
- **RNF03:** Usabilidad: se procurará implementar una interfaz sencilla y de fácil manejo, que sea atractiva para el usuario sin dejar de lado

funcionalidades básicas. El usuario debe encontrar con facilidad las distintas funciones que ofrece la aplicación, y usarlas sin necesidad de formación o de poseer conocimientos avanzados.

- **RNF04:** Fiabilidad: los distintos datos que se guarden deben estar cifrados para garantizar la integridad de los mismos. Los distintos controles de acceso permiten que no haya un comportamiento indebido de un usuario, evitando pérdidas innecesarias de datos.
- **RNF05:** Rendimiento: el rendimiento se verá afectado por el tamaño de los datos que manejará la aplicación, así como el número de usuarios conectados al mismo tiempo. Se pretenderá que ante un número muy grande de ambos, el rendimiento no se vea afectado a través de consultas eficientes a la base de datos y de un código bien estructurado.
- **RNF06:** Escalabilidad: con una buena estructuración de la aplicación, tanto a nivel de código como a nivel de base de datos, se podrán facilitar futuras modificaciones y actualizaciones de la misma.
- **RNF07:** Estabilidad: debido a que se busca un rendimiento eficiente, y a través de unas pruebas completas, no se espera que la aplicación sufra un comportamiento inesperado como una caída del servicio, o un problema en la base de datos, por lo que la estabilidad de la aplicación se buscará que sea constante.

4 Diseño de la solución

El diseño es el proceso de definición de interfaces, arquitecturas, componentes, módulos, pruebas y datos de un sistema software que servirá para satisfacer unos requisitos ya establecidos. En una aplicación web toma especial importancia ya que en esta fase hay que tener en cuenta la navegabilidad, interactividad, usabilidad y la interacción de medios como pueden ser una imagen, video, audio, enlaces, etc.

La programación de la aplicación en su totalidad se ha realizado en el lenguaje PHP. Con ello eliminamos el uso de *frameworks*, que son estructuras que sirven de esqueleto para la propia aplicación, evitando que el programador tenga que plantearse una estructura global.

Teniendo esto en cuenta, la aplicación gana en sencillez, y pasa a tener una estructura muy clara: se creará un fichero PHP por cada página HTML que se quiera crear. Cada página HTML es la que se mostrará al usuario y con la que interactuará. En los ficheros PHP se siguen las mismas pautas de programación, con una inicialización y configuración global, seguido de las conexiones necesarias a la base de datos en las que son necesarias.

Para ello se diferencia el diseño de la interfaz y el diseño de la estructura de datos que se utilizará en la aplicación. Aunque los dos diseños irán conjuntamente en los ficheros PHP de la aplicación, se separan para facilitar su comprensión.

4.1 Diseño de la interfaz

En una interfaz de usuario, el usuario interactúa con un sistema a través de elementos interactivos, menús, imágenes, sonidos, dispositivos de entrada de datos, entre otros.

Una aplicación de estas características, dirigida a un público diverso, requiere de una interfaz y un flujo de navegación intuitivos. Por tanto, buscando la sencillez en esta aplicación, se ha establecido un flujo de navegación que comienza ofreciendo al usuario acceso a la aplicación, siempre que se haya registrado con anterioridad.

En caso contrario también se le ofrece la posibilidad de registrarse, introduciendo los datos necesarios, como son un nombre de usuario, contraseña, dirección de correo electrónico, nombre y número de DNI.

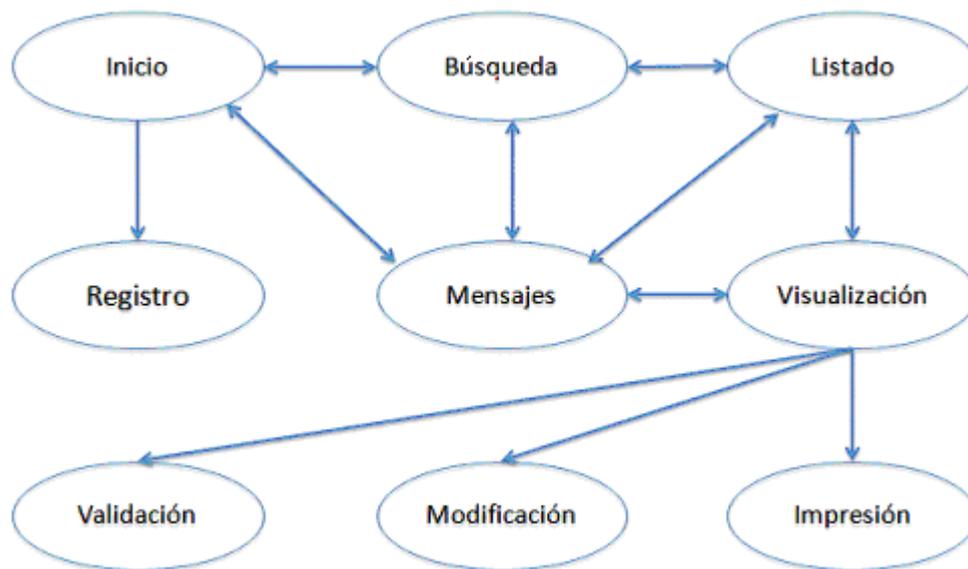


Ilustración 4: Diagrama de Flujo de Navegación

Posteriormente el usuario puede elegir entre realizar una búsqueda entre las asignaturas disponibles o las titulaciones que ofrece la aplicación web. En caso de que quiera que le muestre una lista completa de todas simplemente tiene que dejar en blanco el parámetro de búsqueda. Dicha búsqueda se podrá realizar a través de palabras completas, partes de palabras, o simplemente letras.

Una vez realizada la búsqueda se mostrará un listado con todas las asignaturas / titulaciones que cumplan el parámetro de búsqueda. El usuario podrá elegir cual de ellas quiere visualizar el contenido. En el contenido se mostrará información de la asignatura al igual que la guía docente, dividida en bloques. Habrá un listado de los distintos bloques que contiene la guía docente pudiendo navegar entre ellos mostrando el contenido de cada uno.

Tanto el usuario registrado con privilegios, como un usuario sin haberse registrado, pueden obtener una versión descargable de la guía docente de la asignatura que esta visualizando. Además el usuario registrado puede proceder a crear o modificar un bloque de la guía docente. Este usuario registrado tendrá la de validar una guía docente en la misma página que se visualizan los bloques de la misma y que ofrece la descarga.

En todo momento, después de que el usuario registrado haya accedido a la aplicación, se podrá acceder a los mensajes del usuario en la cabecera de la página, donde se muestra al usuario conectado y el número de mensajes pendientes para leer. Esto mostrará una lista de los mensajes, marcando los que todavía no están leídos. Al pulsar en cualquier de ellos se podrá visualizar su contenido.

4.2 Diseño de la estructura de datos

En una aplicación web se maneja información que tiene que verse almacenada y procesada de alguna manera. Para ello se diseña una estructura de datos destinada a alojar dicha información de la manera más eficiente y segura. Esta estructura de datos utilizará de motor de base de datos MySQL, que permite una rápida lectura de los contenidos que gestiona, así como un funcionamiento multiusuario, es decir, varios usuarios accediendo al mismo tiempo. Como en una aplicación web prima la lectura de datos sobre la modificación de los mismos, MySQL encaja con dichos requisitos, aunque hay que tener en cuenta

que si hubiese una alta concurrencia en la modificación de datos puede haber problemas de integridad.

La base de datos que tendrá toda la información de la aplicación estará localizada en un servidor, y será la aplicación la que acceda al servidor, y por tanto a la base de datos, para recuperar y modificar información. Todas las conexiones a la base de datos se realizarán a través de la extensión de PHP, PDO, que es un controlador que sirve no solo para MySQL, sino también para SQLite, PostgreSQL, entre otras.

El diseño de la base de datos diferencia dos tipos de tablas. Una contiene identificadores y su correspondiente información, y el otro tipo sirve de enlazador entre dos identificadores de distintas tablas.

Las distintas tablas y campos de las mismas que componen la base de datos son los siguientes:

- **Profesores:** guarda los nombres de los profesores que van a componer los distintos equipos docentes de las asignaturas. Además contiene el número de DNI y el correo electrónico. Esta información es actualizada a través de un fichero de configuración.
- **Asignaturas:** guarda los nombres de las asignaturas, con un código único que los identifica y una relación a la titulación que pertenece. Esta información es actualizada a través de un fichero de configuración.
- **Titulaciones:** guarda los nombres de las titulaciones, con un código único que las identifica. Esta información es actualizada a través de un fichero de configuración.
- **Usuarios:** guarda el nombre de usuario, junto con la clave, la dirección de correo electrónico y el número de DNI. Esta información es introducida en la base de datos a través de un formulario PHP, desde la aplicación web.
- **Mensajes:** guarda el título del mensaje, el texto del mismo, la fecha de creación, la asignatura a la que pertenece y el profesor que ha propiciado el mensaje. Esta información es creada a través de la

Virtualización de las Guías Docentes

aplicación cuando un usuario registrado con los privilegios correspondientes realiza un cambio en un bloque de una guía docente.

- **Bloques:** guarda el título del bloque, el texto del mismo, el código de la asignatura a la que pertenece y un código único que lo identifica. Esta información es creada y modificada a través de la aplicación cuando un usuario registrado con los privilegios correspondientes crea o modifica un bloque de una guía docente.
- **ProfAsig:** guarda las relaciones entre los profesores y las asignaturas, y especifica que profesores tienen derechos de coordinador en cada asignatura. Esta información es actualizada a través de un fichero de configuración.
- **ProfMen:** guarda las relaciones entre los profesores y los mensajes, y especifica que profesores han leído un mensaje específicamente. Esta información es actualizada a través de la aplicación cuando un usuario registrado con los privilegios correspondientes realiza un cambio en un bloque de una guía docente, o cuando un usuario registrado lee un mensaje y actualiza el campo leído.

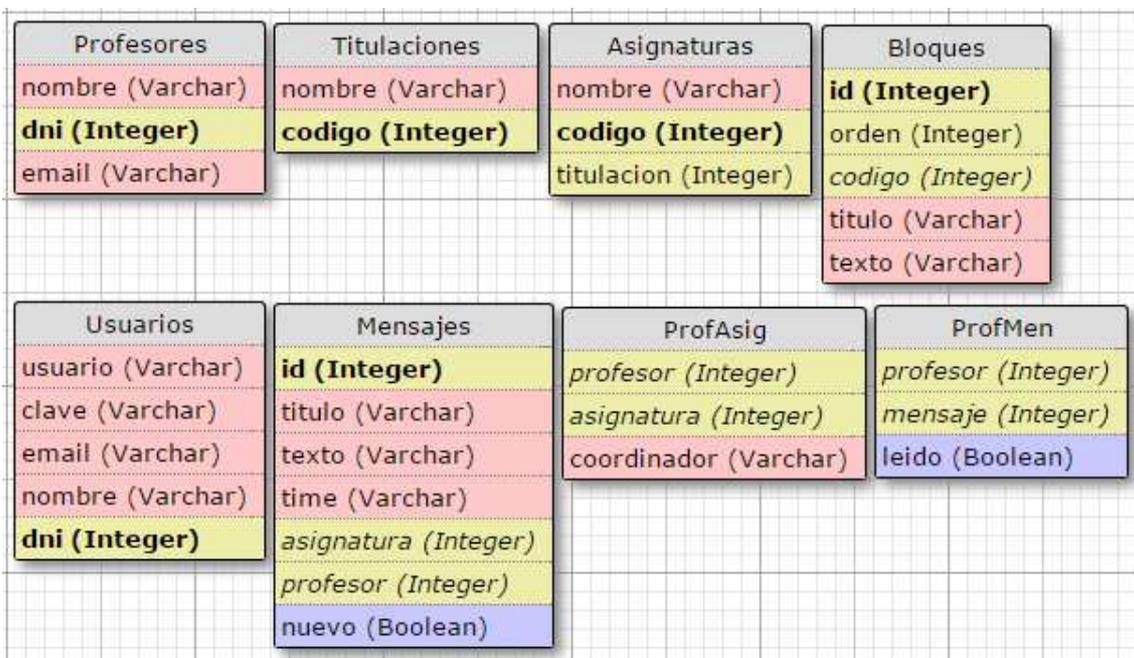


Ilustración 5: Diseño de la Base de Datos

Esta distribución de datos entre las tablas permite transparencia a la hora de acceder a la información que contiene la base de datos. Resaltadas en negrita se encuentran las *PRIMARY KEYS*, que hacen referencia al campo que identifica cada tabla; en cursiva tenemos las *FOREIGN KEYS*, que hacen referencia a una *PRIMARY KEY* de otra tabla para relacionar contenido entre tablas.

Las tablas ProfAsig y ProfMen sirven para relacionar contenido entre las demás tablas para hacer referencia a las tablas profesores / asignaturas y profesores / mensajes respectivamente. Con la primera tabla podemos establecer quien ejerce de coordinador en una asignatura concreta, además de indicar a que asignatura pertenece. Con la segunda tabla indicamos los mensajes que tiene un usuario registrado, indicando si tiene alguno sin leer. Las demás tablas contienen información específica acerca de un campo concreto, como puede ser de un profesor, un bloque de una guía docente, un mensaje, etc.

4.3 Casos de uso

Para poder representar la forma en la que el usuario interactúa con la aplicación se utiliza la técnica de captura de requisitos potenciales de un sistema, es decir, los casos de uso. Los casos de uso nos permite establecer de que forma va a operar el usuario en el sistema, en que orden, o el tipo de elementos que utiliza para ello. En todo caso de uso tenemos los actores, los casos de uso y las relaciones entre ellos.

El siguiente diagrama de casos de uso permite tener una idea general del usuario realizará en la aplicación:

Virtualización de las Guías Docentes

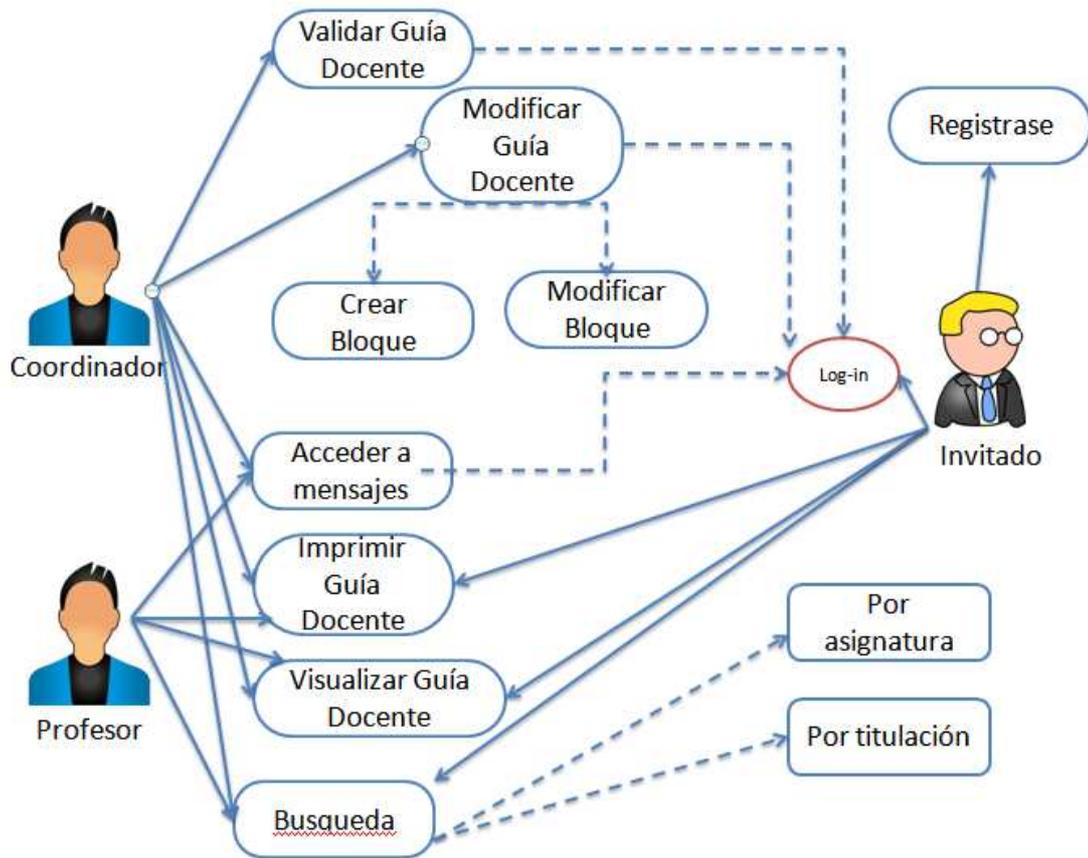


Ilustración 6: Diagrama de Casos de Uso

Los tres tipos de usuario ven identificadas las distintas acciones que pueden realizar con una línea. Hay acciones que requieren de una acción adicional para ser llevadas a cabo, como es el acceso de los mensajes por parte del usuario registrado, que precisa de un Log-In anteriormente. Además hay acciones que se pueden dividir en subacciones ya que pueden ser realizadas de distinta manera, como es el caso de la búsqueda.

El diagrama de casos de uso nos ofrece una versión resumida de los casos de uso de la aplicación, por lo que se procederá a especificar en más detalle los casos de uso que componen la aplicación:

- **CU01: Log-In**

Resumen: Permite a un usuario acceder a la aplicación a través de un usuario y contraseña.

Actores: Coordinador, Profesor.

Precondiciones: el usuario tiene que haberse registrado anteriormente en la aplicación.

Postcondiciones: el usuario accede a la aplicación quedando la sesión abierta. La navegación a través de la aplicación queda marcada con el usuario dentro de la aplicación.

Escenario de éxito:

1. El usuario se encuentra en la página inicial de la aplicación.
2. El usuario introduce el nombre de su usuario y la contraseña en los campos habilitados en la cabecera de la página.
3. El sistema recupera los datos introducidos por el usuario y los comprueba con la base de datos para ver si coinciden.
4. Los datos introducidos por el usuario son verificados por el sistema y son correctos.
5. El sistema registra el acceso del usuario a la aplicación.

Extensiones:

4a: (deriva del punto 4 del escenario de éxito)

1. Los datos introducidos por el usuario, tras ser comprobados por el sistema en la base de datos, son incorrectos.
2. No se produce el acceso del usuario en la aplicación.

Requisitos especiales:

- Comprobación rápida al consultar la base de datos.

Frecuencia de ocurrencia: alta, en función del número de usuarios en la aplicación.

- **CU02: Registrarse**

Resumen: Permite a un usuario registrarse a la aplicación a través de un usuario, una contraseña, una dirección de correo electrónico, un nombre y el número de DNI.

Actores: Invitado.

Precondiciones: el usuario no puede haberse registrado anteriormente en la aplicación.

Postcondiciones: el usuario se registra en la aplicación. Esto permite que el usuario pueda acceder a ella.

Escenario de éxito:

1. El usuario se encuentra en la página inicial de la aplicación.
2. El usuario accede a la página de registro de la aplicación.
3. El usuario introduce un usuario, una contraseña, una dirección de correo electrónico, un nombre y el número de DNI.
4. El sistema recupera los datos introducidos por el usuario y los comprueba con la base de datos para ver si ya existen.
5. Los datos introducidos por el usuario se guardan en la base de datos.
6. El usuario queda registrado en la aplicación.

Extensiones:

5a:

1. Los datos introducidos por el usuario, tras ser comprobados por el sistema en la base de datos, ya existen.
2. No se produce el registro del usuario en la aplicación.

Requisitos especiales:

- Comprobación rápida al consultar la base de datos.
- Modificación sin incongruencias de los datos de las tablas.

Frecuencia de ocurrencia: baja, en función del número de usuarios que quieran registrarse en la aplicación.

- **CU03: Búsqueda de asignaturas/titulaciones**

Resumen: Permite a un usuario realizar una búsqueda en la aplicación de asignaturas y titulaciones a través motor de búsqueda.

Actores: Todos.

Precondiciones: tiene que haber datos en la base de datos para poder realizar la búsqueda.

Postcondiciones: se recibe un resultado en función del parámetro de búsqueda introducido.

Escenario de éxito:

1. El usuario se encuentra en la página inicial de la aplicación.
2. El usuario accede a la página de búsqueda de la aplicación.
3. El usuario introduce un parámetro de búsqueda, ya sea palabra completa o parte de una palabra.
4. El sistema recupera los datos introducidos por el usuario realiza una búsqueda en función del parámetro introducido.
5. Se muestra en la página web una lista de los datos que concuerdan con el parámetro introducido.

Extensiones:

4a:

1. El parámetro introducido por el usuario no concuerda con ningún dato que alberga la base de datos.
2. No se muestra en la página web ningún dato.

Requisitos especiales:

- Comprobación rápida al consultar la base de datos.
- Visualización sin incongruencias de los datos de las tablas.

Frecuencia de ocurrencia: alta, en función del número de usuarios que haya usando la aplicación.

- **CU04: Visualizar Guía Docente**

Resumen: Permite a un usuario visualizar una guía docente almacenada en la base de datos.

Actores: Todos.

Precondiciones: tiene que haber bloques referentes a la guía docente que se quiere visualizar en la base de datos.

Postcondiciones: se muestra los bloques pertenecientes a la guía docente de la asignatura.

Escenario de éxito:

1. El usuario se encuentra en la página inicial de la aplicación.
2. El usuario accede a la página de búsqueda de la aplicación.
3. El usuario introduce un parámetro de búsqueda, ya sea palabra completa o parte de una palabra, para buscar la asignatura concreta.
4. El sistema recupera los datos introducidos por el usuario realiza una búsqueda en función del parámetro introducido.
5. Se muestra en la página web una lista de los datos que concuerdan con el parámetro introducido.
6. El usuario selecciona la asignatura deseada.
7. El sistema muestra una página de la asignatura seleccionada con los bloques de la guía docente.

Extensiones:

4a:

1. El parámetro introducido por el usuario no concuerda con ningún dato que alberga la base de datos.
2. No se muestra en la página web ningún dato.

7a:

1. La asignatura seleccionada no tiene ningún bloque en su guía docente.
2. El sistema no muestra la guía docente de la asignatura.

Requisitos especiales:

- Comprobación rápida al consultar la base de datos.
- Visualización sin incongruencias de los datos de las tablas.

Frecuencia de ocurrencia: alta, en función del número de usuarios que haya usando la aplicación.

- **CU05: Imprimir Guía Docente**

Resumen: Permite a un usuario obtener una guía docente almacenada en la base de datos en formato descargable.

Actores: Todos.

Precondiciones: tiene que haber bloques referentes a la guía docente que se quiere imprimir en la base de datos.

Postcondiciones: se obtiene en formato descargable la guía docente seleccionada.

Escenario de éxito:

1. El usuario se encuentra en la página inicial de la aplicación.
2. El usuario accede a la página de búsqueda de la aplicación.
3. El usuario introduce un parámetro de búsqueda, ya sea palabra completa o parte de una palabra, para buscar la asignatura concreta.
4. El sistema recupera los datos introducidos por el usuario realiza una búsqueda en función del parámetro introducido.
5. Se muestra en la página web una lista de los datos que concuerdan con el parámetro introducido.
6. El usuario selecciona la asignatura deseada.
7. El sistema muestra una página de la asignatura seleccionada con los bloques de la guía docente.
8. El usuario selecciona la opción de obtener en formato descargable la guía docente.
9. El sistema procede a descargar la guía docente en sistema descargable.

Extensiones:

4a:

1. El parámetro introducido por el usuario no concuerda con ningún dato que alberga la base de datos.
2. No se muestra en la página web ningún dato.

7a:

1. La asignatura seleccionada no tiene ningún bloque en su guía docente.
2. El sistema no muestra la guía docente de la asignatura.

9a:

Virtualización de las Guías Docentes

1. La herramienta de conversión a formato descargable produce un error.
2. No se procede a realizar la descarga de la guía docente.

Requisitos especiales:

- Comprobación rápida al consultar la base de datos.
- Visualización sin incongruencias de los datos de las tablas.
- Disponibilidad de la herramienta de conversión

Frecuencia de ocurrencia: media, en función del número de usuarios que haya usando la aplicación.

- **CU06: Leer mensajes**

Resumen: Permite a un usuario registrado leer mensajes que notifican un cambio en una guía docente de una asignatura a la que pertenece.

Actores: Coordinador, Profesores.

Precondiciones: tiene que haber mensajes del usuario almacenados en la base de datos.

Postcondiciones: se procede a leer el mensaje seleccionado por el usuario.

Escenario de éxito:

1. El usuario se encuentra en la página inicial de la aplicación.
2. El usuario introduce el nombre de su usuario y la contraseña en los campos habilitados en la cabecera de la página.
3. El sistema recupera los datos introducidos por el usuario y los comprueba con la base de datos para ver si coinciden.
4. Los datos introducidos por el usuario son verificados por el sistema y son correctos.
5. El sistema registra el acceso del usuario a la aplicación.
6. El usuario accede a la página de mensajes a través de la cabecera.
7. El sistema recupera todos los mensajes del usuario de la base de datos.
8. El sistema muestra una lista de los mensajes en la página web.
9. El sistema muestra el contenido del mensaje en la página web.

Extensiones:

4a:

1. Los datos introducidos por el usuario, tras ser comprobados por el sistema en la base de datos, son incorrectos.
2. No se produce el acceso del usuario en la aplicación.

7a:

1. El usuario no tiene mensajes en la base de datos.
2. El sistema no muestra ningún mensaje en la página web..

Requisitos especiales:

- Comprobación rápida al consultar la base de datos.
- Visualización sin incongruencias de los datos de las tablas.

Frecuencia de ocurrencia: media, en función del número de usuarios registrados que haya usando la aplicación.

- **CU07: Validar Guía Docente**

Resumen: Permite a un usuario registrado con privilegios de la asignatura validar la guía docente de dicha asignatura.

Actores: Coordinador.

Precondiciones: tiene que haber guía docente en la asignatura. Además el usuario tiene que tener derechos de coordinador en dicha asignatura.

Postcondiciones: se procede a validar la guía docente de la asignatura.

Escenario de éxito:

1. El usuario se encuentra en la página inicial de la aplicación.
2. El usuario introduce el nombre de su usuario y la contraseña en los campos habilitados en la cabecera de la página.
3. El sistema recupera los datos introducidos por el usuario y los comprueba con la base de datos para ver si coinciden.
4. Los datos introducidos por el usuario son verificados por el sistema y son correctos.
5. El sistema registra el acceso del usuario a la aplicación.
6. El usuario accede a la página de búsqueda de la aplicación.
7. El usuario introduce un parámetro de búsqueda, ya sea palabra completa o parte de una palabra, para buscar la asignatura concreta.
8. El sistema recupera los datos introducidos por el usuario realiza una búsqueda en función del parámetro introducido.
9. Se muestra en la página web una lista de los datos que concuerdan con el parámetro introducido.
10. El usuario selecciona la asignatura deseada.
11. El sistema muestra una página de la asignatura seleccionada con los bloques de la guía docente.
12. El usuario selecciona la opción de validar guía docente.
13. El sistema comprueba que la guía docente no ha sido modificada contrastándola con la guía docente validada.

Extensiones:

4a:

1. Los datos introducidos por el usuario, tras ser comprobados por el sistema en la base de datos, son incorrectos.
2. No se produce el acceso del usuario en la aplicación.

8a:

1. No existe ninguna asignatura con el parámetro introducido.
2. No se muestra ninguna asignatura en la página web.

11a:

1. La asignatura seleccionada no tiene ningún bloque en su guía docente.
2. El sistema no muestra la guía docente de la asignatura.

12a:

1. La guía docente seleccionada no equivale a la guía docente validada.
2. El sistema muestra un mensaje de error referente al fallo de validación.

Requisitos especiales:

- Comprobación rápida al consultar la base de datos.
- Visualización sin incongruencias de los datos de las tablas.
- Funcionamiento del algoritmo de validación.

Frecuencia de ocurrencia: baja, ocasionalmente un coordinador validará una guía docente.

- **CU08: Modificar Guía Docente**

Resumen: Permite a un usuario registrado con privilegios de la asignatura modificar los bloques de una guía docente de dicha asignatura.

Actores: Coordinador.

Precondiciones: tiene que haber guía docente en la asignatura. Además el usuario tiene que tener derechos de coordinador en dicha asignatura.

Postcondiciones: se procede a modificar el bloque seleccionado de la guía docente de la asignatura.

Escenario de éxito:

1. El usuario se encuentra en la página inicial de la aplicación.
2. El usuario introduce el nombre de su usuario y la contraseña en los campos habilitados en la cabecera de la página.
3. El sistema recupera los datos introducidos por el usuario y los comprueba con la base de datos para ver si coinciden.
4. Los datos introducidos por el usuario son verificados por el sistema y son correctos.
5. El sistema registra el acceso del usuario a la aplicación.
6. El usuario accede a la página de búsqueda de la aplicación.
7. El usuario introduce un parámetro de búsqueda, ya sea palabra completa o parte de una palabra, para buscar la asignatura concreta.
8. El sistema recupera los datos introducidos por el usuario realiza una búsqueda en función del parámetro introducido.
9. Se muestra en la página web una lista de los datos que concuerdan con el parámetro introducido.
10. El usuario selecciona la asignatura deseada.
11. El sistema muestra una página de la asignatura seleccionada con los bloques de la guía docente.
12. El usuario selecciona el bloque que quiere modificar / El usuario selecciona la opción de crear bloque.
13. El sistema redirige al usuario a una página web donde puede introducir el texto del bloque modificado / El sistema redirige al usuario a una página web donde puede introducir el texto del bloque creado, así como el título. El usuario introduce dicha información.

14. El sistema recoge los datos introducidos por el usuario y los guarda en la base de datos.

Extensiones:

4a:

1. Los datos introducidos por el usuario, tras ser comprobados por el sistema en la base de datos, son incorrectos.
2. No se produce el acceso del usuario en la aplicación.

8a:

1. No existe ninguna asignatura con el parámetro introducido.
2. No se muestra ninguna asignatura en la página web.

11a:

1. La asignatura seleccionada no tiene ningún bloque en su guía docente.
2. El sistema no muestra la guía docente de la asignatura.

13a:

1. El sistema no consigue llamar a la herramienta de modificación de texto.
2. El sistema muestra un mensaje de error referente a la herramienta de modificación de texto.

14a:

1. El sistema no puede guardar la información modificada por el usuario.
2. El sistema muestra un mensaje de error referente a la modificación de texto del bloque de la guía docente.

Requisitos especiales:

- Comprobación rápida al consultar la base de datos.
- Visualización sin incongruencias de los datos de las tablas.
- Funcionamiento de la herramienta de modificación de texto

Frecuencia de ocurrencia: media, en función de las asignaturas y usuarios registrados.

5 Implementación

La fase de implementación se aborda cuando, tras haber expuesto los requisitos y diseño de la aplicación, se decide desarrollar la aplicación como tal. En una aplicación web esto supone programar los distintos aspectos de la misma que se han descrito anteriormente, siguiendo un orden y una documentación apropiada. Como se especificó anteriormente, la aplicación esta realizada en PHP para la realización de las páginas HTML a mostrar, y en SQL para las consultas a la base de datos.

5.1 Interfaz de usuario

Para la interfaz de usuario se usan páginas web HTML, codificadas con el lenguaje PHP. Los estilos se especifican en un fichero CSS que permite dar forma a las páginas web HTML, a través de las divisiones creadas en los ficheros PHP.



Ilustración 7: Interfaz de inicio

Dentro de cada PHP encontramos distintos elementos propios del mismo que sirven para implementar la funcionalidad especificada en los requisitos:

- El array asociativo **SESSION**: sirve para establecer sesiones entre ficheros PHP. Es muy útil ya que nos permite intercambiar variables entre páginas HTML, estableciendo `session_start()` al principio de cada fichero. Posteriormente, a través de `$_SESSION["variable"]`, podemos recuperar cualquier valor que hayamos guardado en la sesión con anterioridad. Para cerrar la sesión se utilizan las funciones `session_unset()` y `session_destroy` en ese orden.
- Tratamiento de **formularios**: este elemento de PHP nos permite establecer formularios para la redirección a otras páginas web. Con un campo de tipo *text*, otro de tipo *submit*, y un texto que indique el contenido de la información a rellenar se puede mandar a otro PHP información a través del método POST. Este método de envío es propio de los formularios y la información no se envía a través de la URL, que es la dirección de la página web.
- Uso de funciones de generación de **códigos hash**: para la validación de las guías docentes primero se obtiene en formato descargable, es decir PDF, y posteriormente se procede a comparar el archivo a validar con una copia de la guía docente validada por el coordinador de la asignatura. Para ello se usa la función `sha1_file()` que calcula el código hash sha1 de un archivo específico, y compara dichos códigos. De esta manera se puede comprobar si se ha modificado o falsificado la guía docente.

Además se le añaden librerías externas para poder desarrollar dos funcionalidades más complejas, como son la obtención del fichero descargable de la guía docente y la modificación de los bloques de las guías docentes:

- **NicEdit**: se trata de un editor ligero de plataforma que permite editar texto en una página web de manera sencilla y añadiendo funcionalidades de modificación de texto. Se utiliza como el editor para los textos de las guías docentes.

- **DomPDF:** es un conversor de HTML a PDF. Esto nos permite a partir del contenido de una página web obtener un fichero descargable. Se utiliza para las guías docentes.



Ilustración 8: Editor NicEdit

5.2 Base de datos

Para la base de datos se ha utilizado MySQL, un sistema de gestión de base de datos relacional (basado en lógica de predicados), multihilo (permite tener ejecutándose varias tareas al mismo tiempo) y multiusuario (puede haber varios usuarios conectados a la base de datos al mismo tiempo). Como herramienta que nos permite interactuar con la base de datos se ha usado phpMyAdmin, que permite visualizar todas las bases de datos que tenemos creadas, al igual que las tablas que las contienen. Además nos permite realizar consultas sobre dichas tablas, y modificarlas, sin tener que realizar consultas con SQL.

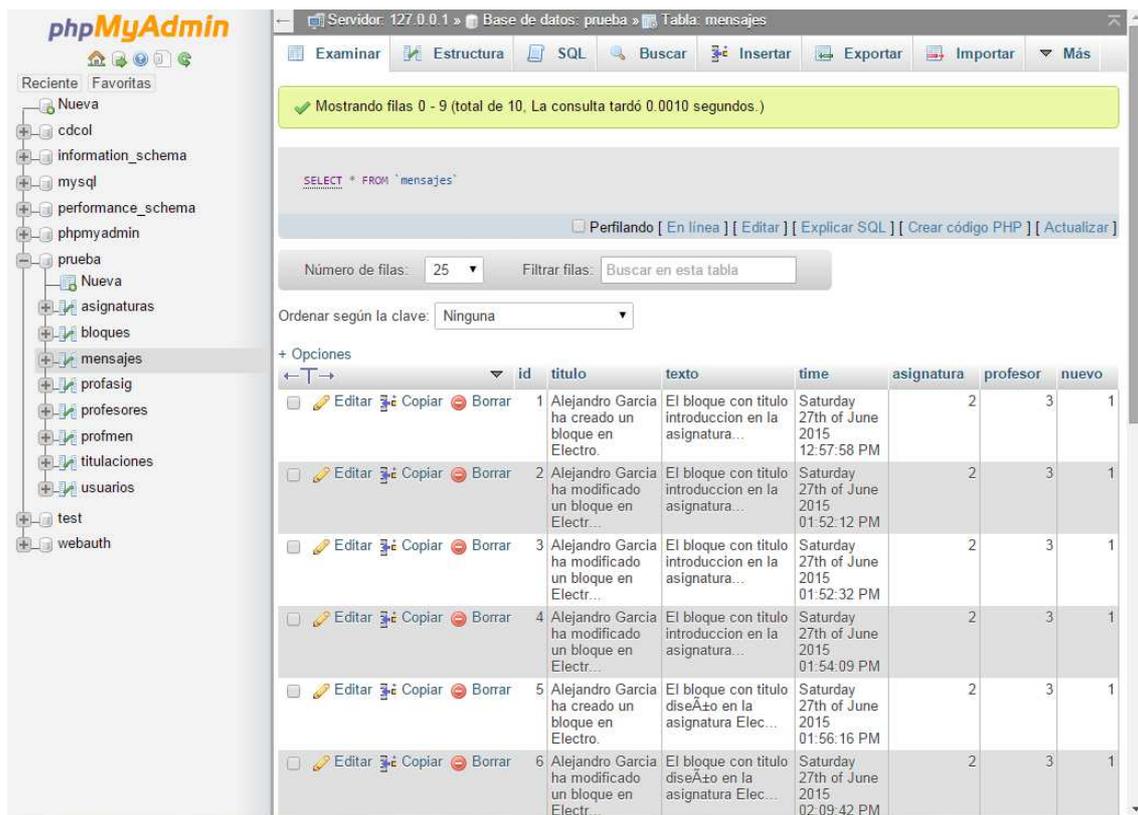


Ilustración 9: Aplicación phpMyAdmin

El primer paso en una base de datos es crear las tablas que la componen. Esto se realiza a través de un fichero, *creatablas.php*, que crea todas las tablas necesarias para poder poblar la base de datos posteriormente. Además realiza un borrado de cualquier tabla que se encontrase anteriormente en ella para evitar incongruencias. Tanto la creación como el borrado de dichas tablas se realizan a través de líneas de código en SQL, donde en la creación de las tablas se usa `CREATE TABLE`, y en el borrado de las mismas se usa `DROP TABLE`.

Como se ha mencionado anteriormente, el poblado de la base de datos inicial se realiza a través de un fichero de configuración, llamado *usuarios.dat*. Dicho fichero contiene información de los profesores, asignaturas y titulaciones que van a componer la base de datos. Además de proporcionar información de cada uno de estos elementos, proporciona la relación que tiene cada dato el uno con el otro, como puede ser que asignatura pertenece a que titulación, que profesor pertenece a que asignatura, etc.

Una vez se tiene el fichero de configuración, se llama a *fichero.php* para que pueda leer la información contenida en el fichero *usuarios.dat*. Para ello se recupera cada línea del fichero y se parte la cadena de caracteres para diferenciar toda la información que contiene. Posteriormente se realizará una consulta SQL que consistirá de un INSERT en las distintas tablas de la base de datos. Se procederá a repetir la misma acción hasta el final del fichero.

A lo largo de la implementación de toda la aplicación se accede con frecuencia a la base de datos, la mayoría de las veces para realizar una consulta de lectura. Esto es el caso de la comprobación de mensajes para un usuario registrado, el acceso a la aplicación por un usuario registrado, visualización de las guías docentes y sus bloques, búsqueda entre las asignaturas y las titulaciones, etc. Pero también se realizan consultas para modificar la información que contiene la base de datos, como se da al registrar un nuevo usuario o modificar los bloques de una guía docente. Para realizar todas estas consultas se necesita establecer una conexión a la base de datos y para ellos se utiliza el controlador PDO, que se trata de una extensión de PHP para realizar dichas conexiones. No es específico de MySQL pero si que proporciona todas las herramientas necesarias para la aplicación.

5.3 Funcionalidad

Una vez se ha tenido en cuenta la implementación de la interfaz y la base de datos, se implementa la funcionalidad que se ha descrito de la aplicación web. Teniendo en cuenta los distintos usuarios que pueden usar esta aplicación, se definen las distintas acciones que puede realizar cada tipo de usuario:

<u>Tarea</u>	<u>Frecuencia</u>	<u>Usuarios</u>	<u>Entradas</u>	<u>Salidas</u>	<u>Posibles Problemas</u>
<i>Registrarse</i>	1	Invitado	Usuario Contraseña Email Nombre DNI	Devuelve la confirmación del registro del usuario.	Introducción de datos no válidos. El usuario ya se había registrado.
<i>Log-In</i>	3	Coordinador Profesor	Usuario Contraseña	Devuelve autenticación del usuario.	Introducción de datos incorrectos.
<i>Búsqueda</i>	4	Todos	Parámetro de búsqueda	Devuelve una lista de información que encaja con el parámetro.	No se encuentra ningún dato con dicho parámetro.
<i>Visualizar Guía Docente</i>	4	Todos	-	-	No existe guía docente.
<i>Modificar Guía Docente</i>	2	Coordinador	Texto	Devuelve una confirmación de la modificación.	Error al modificar el texto en la base de datos.
<i>Validar Guía Docente</i>	1	Coordinador	-	-	No existe guía docente.
<i>Acceder a mensajes</i>	3	Coordinador Profesor	-	Devuelve la lista de mensajes del usuario.	No existen mensajes.
<i>Imprimir Guía Docente</i>	2	Todos	-	Devuelve en formato descargable la guía docente	No existe guía docente.

Virtualización de las Guías Docentes

En la visualización de la guía docente se puede acceder a los distintos bloques que la componen, a través de un índice que se sitúa debajo de la información del profesorado. Con dicho índice se navega entre el contenido de los bloques, pudiendo modificarlos o crear nuevos si se dispone de permisos para ello. Por último se podrá tanto imprimir la guía docente en su totalidad, como comprobar que la guía docente de la que disponemos es la misma que se encuentra almacenada en la aplicación, para evitar falsificaciones y uso indebido de las mismas.

Escuela Politécnica Superior

Logueado como pedro(5) Salir

Inicio Listado Busqueda UAM Edición de Guías Docentes

Coordinador:
Pedro Gomez

Profesores:
Jose Ignacio Ruiz

Redes de Comunicacion

Bloques de la Guia Docente: 7

- [Datos del equipo docente](#)
- [Objetivos del Curso](#)
- [Contenidos del Programa](#)
- [Metodos Docentes](#)
- [Metodos de Evaluacion](#)
- [Cronograma](#)
- [Referencias de Consulta](#)

-Titulo del bloque actual:

Datos del equipo docente

-Contenido del bloque actual:

Dr. Pedro Gomez
Departamento de Tecnologia Electronica y Comunicaciones
Escuela Politecnica Superior
Despacho: C-513 Edificio D 2º Planta
Correo Electronico: pedro@ejemplo.es

Dr. Jose Ignacio Ruiz
Departamento de Tecnologia Electronica y Comunicaciones
Escuela Politecnica Superior
Despacho: C-241 Edificio A 1º Planta
Correo Electronico: jose@ejemplo.es

Editar Bloque Actual Nuevo Bloque Validar PDF Imprimir PDF

Ilustración 10: Visualización de la guía docente

6 Pruebas y resultados

La fase de pruebas es el proceso de someter la aplicación web a una serie de actividades que permita descubrir errores dentro de la misma. Con esto se persigue asegurar el correcto funcionamiento de la aplicación y corroborar que se han seguido las pautas especificadas en la fase de análisis de requisitos.

Se realizará una batería de pruebas preestablecida, desde las pruebas que tienen en cuenta los componentes más pequeños de la aplicación, hasta las pruebas que consideran la aplicación entera como un todo. Es por ello por lo que se tiene que seguir un orden cuando se realizan las pruebas. Se verán diferenciadas en distintos tipos:

- **Pruebas unitarias:** comprueban la lógica, funcionalidad y especificación de cada módulo independientemente. Se llevan a cabo en la fase de implementación en su mayoría.
- **Pruebas de integración:** consisten en agrupar módulos y comprobar el funcionamiento conjunto entre ellos. Casi exclusivamente se llevan a cabo en la fase de implementación.
- **Pruebas de sistema:** tienen en cuenta la integración de la aplicación en su entorno hardware y software. Estas pruebas son llevadas a cabo durante todo el transcurso del proyecto.
- **Pruebas de validación:** sirven para verificar que la aplicación cumple con las especificaciones y requisitos marcados en fases anteriores. Normalmente llevadas a cabo al terminar cada iteración de la fase de implementación.

Además de los tipos de pruebas mencionadas, se pueden diferenciar según el conocimiento interno del módulo que se está probando:

- **Pruebas de caja blanca:** en ellas se conoce el código fuente y las pruebas son específicas para ese módulo. Esto implica que cualquier cambio en el código haría que las pruebas fuesen modificadas de la misma manera. Al tratar con el código fuente directamente requiere

mayores conocimientos técnicos y una inversión más grande de tiempo.

- **Pruebas de caja negra:** en ellas se conoce la funcionalidad del módulo en cuestión pero no su código. Solo es necesario conocer que entradas necesita para su funcionamiento y las salidas correctas para comprobar su funcionamiento. Debido a esto el conocimiento técnico que se requiere para realizarlas es menor.

Una vez descritas las pruebas llevadas a cabo en términos generales, se profundizará más en ellas, dando ejemplos que ilustran como se han realizado y con que criterios.

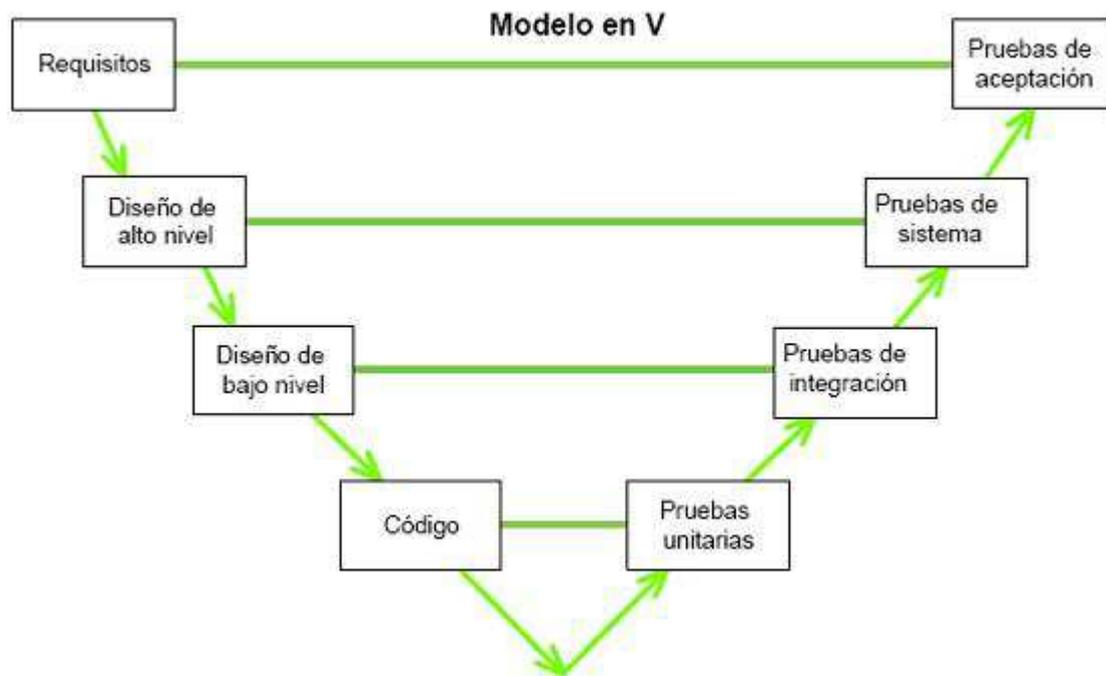


Ilustración 11: Tipos de pruebas realizadas en el proyecto

6.1 Pruebas unitarias

Las pruebas unitarias son las primeras que se realizan sobre un proyecto. Esto es debido a que nos permiten probar pequeños fragmentos de código denominados módulos y a medida que avanzan las pruebas crear módulos que

contengan los anteriores. La realización de estas pruebas suelen darse en la fase de implementación y facilita la localización de errores ya que se pueden descartar módulos ya probados frente a módulos más grandes. Las pruebas unitarias utilizadas son pruebas de caja blanca, comprobando el funcionamiento de los módulos a nivel de código fuente. Algunos ejemplos de las pruebas unitarias realizadas son:

- **PU01: Acceso a la base de datos y modificación de las tablas**

En el fichero creatablas.php accedemos a la base de datos para proceder a la creación de las tablas que van a utilizarse en la aplicación. Antes de crear dichas tablas se realiza un borrado completo de lo que contiene la base de datos, tanto de las tablas como de la información que contienen. Posteriormente se procede a la creación de las tablas con sus campos respectivamente. Para verificar que se realiza todo correctamente se tiene en cuenta lo siguiente:

- La conexión a la base de datos no devuelve ningún error.
- Se visualiza la base de datos a través de la interfaz phpMyAdmin para comprobar si existe algún contenido antes de ejecutar el fichero.
- Se comprueba que se crean las tablas mencionadas a través de la interfaz phpMyAdmin y que se encuentran con sus respectivos campos sin contenido de información.

- **PU02: Visualización de contenido de la base de datos**

Al realizar una consulta SQL a la base de datos se obtiene un resultado. Dicho resultado es recogido en PHP a través de la extensión PDO. Antes de realizar las consultas necesarias para implementar la funcionalidad de la aplicación hay que comprobar con consultas simples que el resultado de las consultas es correcto. Para ello hay tener en cuenta lo siguiente:

- El código PHP no devuelve ningún error al realizar la conexión PDO a la base de datos seleccionada.
- Al realizar la consulta no se devuelve ningún error de sintaxis SQL.

- Cuando se obtiene el resultado no hay incongruencias y la consulta es satisfactoria.

- **PU03: Uso de la herramienta DomPDF**

Para poder obtener una copia de la guía docente en formato descargable, en este caso PDF, se usa la herramienta DomPDF. Esta herramienta transforma un texto guardado en una variable PHP en un fichero PDF. Para ello se renderiza dicho texto y se especifica el directorio de salida. Para verificar el correcto funcionamiento se tiene en cuenta lo siguiente:

- No hay fallo al incluir la librería de DomPDF.
- No muestra error de renderización del texto de prueba por parte de la herramienta.
- Se descarga el fichero en formato PDF y contiene el texto sin erratas.

6.2 Pruebas de integración

Una vez aprobadas las pruebas unitarias, se procederá a realizar las pruebas de integración. En las pruebas unitarias se comprobaba el funcionamiento de los módulos de forma independiente. En las pruebas de integración se determina que módulos se combinan para proceder a comprobar su funcionamiento de forma conjunta. Se utilizan pruebas de caja blanca en mayor escala que las pruebas unitarias ya que los módulos son más grandes. Estas pruebas son eliminadas de la aplicación posteriormente, de igual manera que las unitarias, ya que la información que despliegan es solo de utilidad para el programador. Algunos ejemplos de las pruebas de integración realizadas son:

- **PI01: Búsqueda de asignaturas/titulaciones.**

Para la realización de búsquedas en la aplicación se junta el lenguaje PHP, que actúa de interfaz con el usuario, y el lenguaje SQL, que sirve para acceder a la base de datos y recuperar la información solicitada.

Para realizar una búsqueda el usuario tiene que introducir en un área de texto el parámetro de búsqueda, ya sea una palabra o simplemente letras que puedan contener la asignatura o titulación que buscamos. A través de PHP utilizamos ese parámetro para realizar una consulta SQL a la base de datos. Todo esto se tiene en cuenta en la prueba verificando lo siguiente:

- El fichero PHP muestra correctamente la página HTML.
- La información mostrada después de la consulta encaja con el parámetro introducido por el usuario.
- No existe ninguna incongruencia con la información desplegada tras la consulta a la base de datos.

- **PI02: Uso del editor NicEdit**

La inclusión de la librería externa NicEdit nos permite añadir funcionalidad a la hora de modificar los bloques de la guía docente. Se añade al principio del fichero PHP, en el que se quiere usar la herramienta, la referencia a dicha librería y cualquier elemento PHP de tipo *textarea* pasará a tener el formato de NicEdit. Para comprobar el correcto funcionamiento se tiene que dar las siguientes condiciones:

- No hay fallo al referenciar la librería.
- El fichero PHP conecta correctamente con la base de datos.
- La página web muestra correctamente la caja de herramientas del editor.

- **PI03: Impresión guías docentes**

Habiendo probado independientemente la herramienta DomPDF, la impresión de las guías docentes en formato viene determinado por los bloques que almacena la base de datos con respecto a la asignatura. Para ello se necesita a través de PHP realizar una consulta que recupere los bloques y posteriormente guardarlos en una variable. Con ello se puede renderizar la información y proceder a la impresión. Para verificar el correcto funcionamiento se tiene en cuenta lo siguiente:

- No hay fallo al incluir la librería de DomPDF.

- No muestra error de renderización del texto posterior a la consulta por parte de la herramienta.
- Se descarga el fichero en formato PDF y contiene el texto sin erratas.

6.3 Pruebas de sistema

Las pruebas de sistema se han realizado constantemente a lo largo del desarrollo del proyecto. Se ha trabajado desde dos máquinas distintas que han albergado localmente tanto el servidor como la aplicación. Dichas máquinas utilizan el sistema operativo Windows 7 y Windows 8. Además la aplicación ha sido desplegada en los navegadores Internet Explorer, Google Chrome, y Mozilla Firefox en sus últimas versiones.

Durante las pruebas de sistema no ha surgido problema alguno en ninguno de los entornos mencionados. Esto es debido a que a la aplicación no se le aplica ninguna restricción que pueda ofrecer limitaciones a un entorno concreto. Con la diversidad de hardware y software que han supuesto el entorno de la aplicación en la que se ha probado, se puede confirmar el correcto funcionamiento en cualquier otro entorno hardware y software.

6.4 Pruebas de validación

Las pruebas de validación tienen como meta comprobar que el producto final cumple con los requisitos que se fijaron en la fase de análisis. Los requisitos que se fijaron pretendían ser esenciales para definir la funcionalidad de la aplicación y poder darle forma. Por ello a pesar de haber probado cada una de las funcionalidades independientemente a medida que se iba avanzando en el proyecto, se necesitan hacer pruebas de caja negra una vez acabada la fase de implementación y posterior a las pruebas unitarias, de integración y de

sistema, para asegurarse que se han complicado los requisitos y opinar en función de los resultados.

Estas pruebas de caja negra consisten en introducir unos parámetros de prueba a la aplicación simulando un usuario accediendo a las funcionalidades que ofrece. Con esto se pretende obtener una perspectiva lo más real posible de si se ha alcanzado lo que se proponía. En el caso de un proyecto encargado por un cliente, estas pruebas servirían para responder a la pregunta, ¿es esto lo que el cliente quiere?

6.5 Otras pruebas

Se podrían haber considerado otro tipo de pruebas para comprobar el funcionamiento de la aplicación en cuestión pero se descartaron debido a que podrían resultar repetitivas, redundantes, o que por el tamaño del proyecto resultarán insignificantes. Entre estas pruebas se encuentran:

- **Pruebas de interfaz:** vienen definidas por la usabilidad y suelen realizarse en una fase inicial del proyecto. Debido a la modularidad que se aplicó en las pruebas unitarias se consideraron innecesarias.
- **Pruebas de navegación:** comprueban el correcto funcionamiento de la aplicación a nivel estético, comprobando que no existen enlaces rotos, o botones que no tienen ninguna funcionalidad. Debido al tamaño de la aplicación se consideró relevante.
- **Pruebas de seguridad:** aunque resultan ser una obligación en casi todos los proyectos software, el hecho de que la aplicación no centrarse en ningún momento una prioridad en la seguridad ha propiciado no considerar este tipo de pruebas. Sí es cierto que debería considerarse a la hora de realizar mejoras, tanto para proteger los datos sensibles de la base de datos, como el proceso de validación de las guías docentes.
- **Pruebas de configuración:** se realizan cuando el proyecto se pretende desplegarlo en un sistema concreto, por lo que tiene que

cumplir con unas características concretas y ceñirse a ellas. En este proyecto no es el caso.

- **Pruebas de base de datos:** al igual que en las pruebas de interfaz, las pruebas unitarias abarcan de forma genérica este tipo de pruebas, por lo que añadirlas sería redundante.

6.6 Resultados

Los resultados que se obtienen a partir de las pruebas mencionadas son óptimos. El número de errores, en concreto los críticos, se han ido reduciendo de forma drástica a medida que avanzaba el proyecto, y el ciclo de vida ha permitido una resolución rápida y con un impacto mínimo.

El proyecto al estar en continua evolución ha permitido que el producto final cumpla las expectativas fijadas desde un principio. Las pruebas unitarias, de integración y de sistema permiten certificar que el funcionamiento de la aplicación será el correcto, incluso si se desplegara en otro entorno distinto al probado durante las pruebas, debido a la capacidad que tiene dicha aplicación de adaptarse y mantener su funcionalidad intacta.

Por último las pruebas de validación permiten confirmar que los requisitos fijados en la fase de análisis se han cumplido y que el producto final es satisfactorio en lo que a este campo respecta. Remarca que habría que tener en cuenta otro tipo de pruebas quizá más exhaustivas si se procediese a ampliar la aplicación en algunos aspectos.

7 Conclusiones y trabajo futuro

El Trabajo Fin de Grado realizado ha estado guiado fundamentalmente por una componente de desarrollo acorde con algunas de las competencias desarrolladas a lo largo del Grado en Ingeniería Informática, del que este trabajo es la conclusión.

Desde el punto de vista de los objetivos fijados al comienzo de este proyecto, se puede concluir que estos se han cumplido en su totalidad. Se fijó como meta la realización de una aplicación web para solucionar el problema que resultaba de la administración de las guías docentes. La aplicación desarrollada establece unas funcionalidades que permiten facilitar la tarea y añadir componentes relevantes a la hora de considerar la creación y modificación de una guía docente.

Se han aplicado conocimientos adquiridos a lo largo de la carrera para poder realizar este proyecto de la manera más eficiente posible. Estos conocimientos resultaron ser suficientes, en cierta medida, a la hora de emprender un proyecto de esta envergadura. En los momentos en que los conocimientos no fueron los necesarios y a medida que la dificultad del proyecto creció, la madurez adquirida en el transcurso de los estudios de grado me permitió ampliar los conocimientos, dando paso al aprendizaje necesario para poder abarcar todas las funcionalidades que necesitaban ser desarrolladas.

Como opinión personal, me ha resultado un proyecto muy interesante que, aplicado a un problema real, me ha hecho ver la importancia de las aplicaciones web en el entorno laboral de hoy en día. En un espacio tan amplio como es Internet, tener conocimientos de cómo funcionan las páginas web y las aplicaciones web resulta ser algo esencial para poder desenvolverse en este entorno.

7.1 Trabajo futuro

Todo proyecto de envergadura, como puede ser el que se ha presentado en esta memoria, siempre tiene aspectos que mejorar. En nuestro caso, a pesar de que se ha realizado el proyecto atendiendo a las fases de vida, han quedado aspectos sin resolver o se ha comprobado a posteriori que sería preferible haber enfocado ciertas decisiones desde otro punto de vista. En este sentido cabría destacar:

- El manejo y estructuración de la base de datos. Aunque la implementación llevada a cabo en este proyecto fuese, desde mi punto de vista, la más adecuada, si es verdad que una ampliación de la aplicación en cuanto a componentes de la misma, haría que la base de datos mostrase carencias y necesitara una reestructuración.
- La ausencia de frameworks puede resultar desde un primer momento cómodo durante el desarrollo de la aplicación. Al ser una aplicación limitada no se consideró oportuno implementarlos. Pero con la inclusión de mejoras futuras, la aplicación podría llegar a tener un tamaño en el que se podría considerar el uso de frameworks, tanto creados por el usuario, como algunos ya desarrollados por un tercero, pudiendo implementar el modelo vista controlador.
- La validación de las guías docentes. Las opciones son múltiples, se optó por una relativamente sencilla pero que cumpliera su cometido, pero cabe destacar que hay maneras más elegantes para tratar el problema, desde utilizar algoritmos más seguros de codificación, a mejorar la interfaz de validación de archivos.

Bibliografía

Libros:

- Ware, Brent, et al. Open Source Development with LAMP: Using Linux, Apache, MySQL and PHP. Addison-Wesley Longman Publishing Co., Inc., 2002.
- Bakken, Stig Sæther, et al. Manual de PHP. Este manual es© Copyright 1997, 1998, 1999, 2000, 2001 del Grupo de documentación de PHP. 1997.
- MYSQL, A. B. MySQL. 2001
- Roldán Martínez, David; Valderas Aranda, Pedro J; Pastor López, Óscar 2010: Aplicaciones web, un enfoque práctico.
- Moseley, Ralph Manual avanzado, 2007: Manual avanzado de desarrollo de aplicaciones web.

Links:

- DomPDF.
<https://code.google.com/p/dompdf/>
(accedido por última vez el 01/07/2015)
- Ayuda DomPDF.
https://manuais.iessanclemente.net/index.php/Generaci%C3%B3n_de_PDF's_con_PHP_y_la_librer%C3%ADa_DOMPDF
(accedido por última vez el 01/07/2015)
- NicEdit.
<http://nicedit.com/>
(accedido por última vez el 01/07/2015)
- Ayuda NicEdit.
<http://wiki.nicedit.com/w/page/519/FrontPage>
(accedido por última vez el 01/07/2015)
- Dudas varias.
<http://stackoverflow.com/>
(accedido por última vez el 01/07/2015)

- Ayuda con las definiciones.
<https://es.wikipedia.org/>
(accedido por última vez el 01/07/2015)
- phpMyAdmin.
http://www.phpmyadmin.net/home_page/index.php
(accedido por última vez el 01/07/2015)
- XAMPP.
<https://www.apachefriends.org/es/index.html>
(accedido por última vez el 01/07/2015)
- Tecnologías empleadas.
<http://www.w3schools.com/>
(accedido por última vez el 01/07/2015)
- Imágenes varias.
aplicacioneswebjag08380865.blogspot.com
(accedido por última vez el 01/07/2015)

Glosario

Aplicación web: aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web a través de Internet mediante un navegador.

Base de Datos: un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.

Bloque de una guía docente: sección que describe una parte de la guía docente.

Ciclo de vida: estructura aplicada al desarrollo de un producto de software.

Enlace roto: Dirección de una página web la cual no conduce a ningún sitio.

Feedback: comentarios que se reciben de un tercero sobre un tema en concreto. Retroalimentación.

Framework: Estructura utilizada para el desarrollo de una aplicación software.

Foreign Key: Hace referencia a una Primary Key de otra tabla de la base de datos.

Guía docente: instrumento que sirve de documento de planificación de toda la actividad docente.

HTTP: Protocolo que se usa en cada transacción de Internet.

Moodle: aplicación Web que sirve de plataforma para intercambio de recursos e información entre estudiantes y equipo docente.

Plataforma: un sistema que sirve como base para hacer funcionar determinados módulos de hardware o de software con los que es compatible.

Primary Key: Define el campo principal y único de una tabla de la base de datos.

Recurso: medio de cualquier clase que, en caso de necesidad, sirve para conseguir lo que se pretende.

Requisito: se trata de una descripción completa del comportamiento del sistema que se va a desarrollar.

Usuario: aquella persona que hace uso de un sistema software.

UAM: Universidad Autónoma de Madrid.

EPS: Escuela Politécnica Superior.

TFG: Trabajo de Fin de Grado.

EEES: Espacio Europeo de Educación Superior.

TIC: Tecnologías de la Información y las Comunicaciones.