



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escola Tècnica
Superior d'Enginyeria
Informàtica

Proyecto Final de Carrera

Aplicación Web para supervisar la asistencia a las sesiones de prácticas

Autor

Abel Llopis Granero

Director

Sergio Saez Barona

Titulación

Ingeniería técnica informática de gestión

Fecha

16 de enero de 2012

1. INTRODUCCIÓN	5
1.1. MOTIVACIÓN	6
1.2. SITUACIÓN GENERAL	6
1.3. OBJETIVOS	6
2. ESPECIFICACIÓN DE REQUISITOS	8
2.1 INTRODUCCIÓN	9
2.1.1 OBJETIVO	9
2.1.2 ÁMBITO DEL SISTEMA	9
2.1.3 DEFINICIONES, ACRÓNIMOS Y ABREVIACIONES	9
2.2 DESCRIPCIÓN GENERAL	10
2.2.1 PERSPECTIVA DE LA APLICACIÓN	10
2.2.2 FUNCIONES DE LA APLICACIÓN	10
2.2.3 CARACTERÍSTICAS DEL USUARIO	11
2.2.4 RESTRICCIONES GENERALES	11
2.3 REQUISITOS ESPECÍFICOS	11
2.3.1 REQUISITOS FUNCIONALES	11
2.3.2 REQUISITOS DE INTERFACES EXTERNAS	13
2.3.3 REQUISITOS DE EFICIENCIA	14
2.3.4 RESTRICCIONES DE DISEÑO	14
2.3.5 ATRIBUTOS	14
2.3.6 OTROS REQUISITOS	15
3. ANÁLISIS	16
3.1 CASOS DE USO	17
3.1.1 ACTORES	17
3.1.2 ESCENARIOS	18
4. DISEÑO	24
4.1 ARQUITECTURA SOFTWARE	25
4.1.1 CAPA DE PRESENTACIÓN	25
4.1.2 CAPA DE NEGOCIO O LÓGICA DE CONTROL	27
4.1.3 CAPA DE PERSISTENCIA	28
5. IMPLEMENTACIÓN	31
5.1 TECNOLOGÍA	32
5.2 METODOLOGÍA	32
5.3 ESTRUCTURA DE LA APLICACIÓN	32
6. PRUEBAS DE USO	34
6.1 PRUEBAS REALIZADAS	35
6.1.1 PRUEBAS FUNCIONALES	35
6.1.2 PRUEBAS DE VISUALIZACIÓN	36
6.1.3 PRUEBAS DE TIEMPO DE EJECUCIÓN	38
6.1.4 PRUEBAS DE DUPLICIDAD DE CÓDIGO	38
6.1.5 PRUEBAS DE USO	38
7. CONCLUSIONES	44
7.1 OBJETIVOS CUMPLIDOS	45
7.2 AMPLIACIONES	45
7.3 CONCLUSIÓN PERSONAL	45

8.	BIBLIOGRAFÍA	47
8.1	LIBROS	48
8.2	SITIOS WEB DE REFERENCIA	48
9.	ANEXO A	49
9.1	INSTALACIÓN	50
9.2	CONFIGURACIÓN	50
9.2.1	AULAS	51
9.2.2	PRÁCTICAS	53

1. Introducción

Describimos la motivación para la realización del proyecto así como la necesidad de su desarrollo y los objetivos generales que este debe cumplir.

Comprenderemos la finalidad de la realización del proyecto y que queremos conseguir con su desarrollo.

1.1. Motivación

A medida que pasa el tiempo, la tecnología avanza y para ello, la sociedad debe de ir adaptándose a ella, es por esto que se intenta agilizar y optimizar los procedimientos que pueden mejorarse para que se realicen de la forma más cómoda y rápida posible.

La motivación principal para el desarrollo de una aplicación web para la gestión de la asistencia y evaluación de las sesiones prácticas, es la posibilidad de poder crear un sistema que permita controlar la presencia del alumno a las prácticas de una manera cómoda, rápida y segura.

1.2. Situación general

Actualmente el profesor debe pasar lista, leyendo uno a uno cada alumno para comprobar que está en la sesión de prácticas a la que le corresponde y que no hay ningún alumno que pertenece a otro grupo.

Esto hace que se consuma un tiempo de la sesión de prácticas para realizar el control de los alumnos asistentes y reduce el tiempo que tienen los alumnos para realizar los ejercicios de las prácticas.

También tiene el inconveniente que si se desea conocer si un alumno realizó una práctica hace unas semanas, debería buscar las hojas de las prácticas entregadas para conocer si el alumno las presentó sus ejercicios y la nota que este obtuvo.

En este ambiente, existe el riesgo que se puedan traspapelar las listas de asistentes o las listas de las prácticas entregadas, es por ello que requiere una organización por parte del profesor así como una planificación para almacenar la documentación y que esta no se extravíe.

1.3. Objetivos

El objetivo principal del proyecto es el desarrollo de una aplicación web para gestionar la asistencia y evaluar las sesiones de prácticas de una manera rápida, fiable y con la que se pueda optimizar el tiempo de las sesiones.

De esta manera los alumnos dispondrán de la totalidad del tiempo para realizar los ejercicios correspondientes a cada sesión, consultar si realizaron todas las prácticas y que nota obtuvieron.

Por parte del profesor, podrá evaluar a los alumnos asignando las prácticas como entregadas o no y si asistieron a una sesión de prácticas concreta.

Con todo ello se intenta agilizar el control de las sesiones de prácticas. El alumno puede conocer si realizó una práctica y conocer su estado. Esto le

ayudará a planificarse la evaluación y conocer que prácticas le quedan y que le queda por hacer.

En la parte que incumbe al profesor, la implantación de este sistema podrá librarlo de varias tareas que aunque siendo tareas que deben realizar, con simplemente acceder al menú de la aplicación podrán conocer quién ha asistido a clase y que prácticas ha realizado, así como saber si algún alumno no está en su aula correcta.

2. Especificación de requisitos

En esta sección detallamos los objetivos de la aplicación, las funciones que debe realizar y las necesidades que debe cubrir, así como las restricciones que contiene.

2.1 Introducción

El presente documento se elabora a los efectos de servir como punto de referencia en cuanto a la especificación de requisitos de la aplicación y para que forme parte de la documentación de la aplicación.

2.1.1 Objetivo

El objetivo principal de la especificación de requisitos es definir, organizar y documentar las funcionalidades que cubrirá nuestro sistema, de manera que se cumpla tanto las expectativas del cliente como la de los futuros usuarios.

2.1.2 *Ámbito del sistema*

La aplicación web para la gestión de la asistencia y evaluación de sesiones de prácticas contendrá las siguientes funcionalidades:

- Registro de ingreso de los alumnos a la sesión de prácticas
- Registro del profesor a la sesión de prácticas.
- Registro de las prácticas entregadas por los alumnos.
- Validación de la entrega de las prácticas de los alumnos.

2.1.3 *Definiciones, acrónimos y abreviaciones*

Esta sección pretende servir como punto de referencia a una persona no vinculada al área informática y familiarizarlo con los términos y conceptos utilizados.

login

Es el momento de identificación al ingresar en el sistema.

logout

Es el momento de cerrar la sesión en el sistema

HTML

HyperText Markup Language, Lenguaje de marcado de hipertexto, es el lenguaje para la programación de páginas web.

PHP

PHP Hypertext Pre-processor. Es el lenguaje de programación para la creación de páginas web dinámicas.

MySQL

Es un sistema de gestión de bases de datos.

XML

Extensible Markup Language. Es un lenguaje basado en etiquetas para compartir información de una manera segura, fiable y fácil.

2.2 Descripción general

A continuación se describirán todos aquellos factores que afectan a la aplicación y a sus requisitos.

Se detallaran las funciones de la aplicación y que usuarios pueden realizarlas.

2.2.1 Perspectiva de la aplicación

Ante la decisión de crear un registro de los alumnos que acuden y realizan las sesiones de prácticas, se plantea la realización de una aplicación web para la gestión de la asistencia y evaluación de sesiones de prácticas que permite la comprobación rápida y ágil de los asistentes a una sesión.

La aplicación funcionará bajo cualquiera de los sistemas operativos del mercado, ya que la aplicación se ejecuta sobre un navegador web por lo que el único requisito para funcionar es poseer un navegador web para ejecutar la aplicación.

2.2.2 Funciones de la aplicación

Las funciones de la aplicación se pueden separar en 2 bloques dependiendo del usuario que acceda a ella.

a) Alumno:

- Login en el sistema
- Login en una práctica
- Consultar información personal
- Consultar prácticas realizadas
- Consultar prácticas entregadas
- Consultar histórico de sesiones
- Logout en el sistema

b) Profesor:

- Login en el sistema
- Establecer una práctica para la sesión actual
- Consultar información personal
- Consultar el mapa de la clase

- Consultar prácticas realizadas por cada alumno
- Marcar las prácticas de los alumnos como entregadas
- Consultar el histórico de sesiones
- Logout de un alumno, de un grupo o de todos los alumnos.
- Logout en el sistema

2.2.3 Características del usuario

Existen 2 tipos de usuarios:

- Alumnos: Alumnos de la UPV registrados en el sistema
- Profesores: Profesores de la UPV registrados en el sistema.

2.2.4 Restricciones generales

La aplicación está destinada al uso en las aulas de prácticas de la UPV, funciona con un navegador web por lo que es necesario que el sistema operativo disponga de uno.

2.3 Requisitos Específicos

2.3.1 Requisitos funcionales

Definiremos las acciones que debe realizar el software, es decir, de qué forma se procesan y aceptan un conjunto de entradas para producir una determinada salida.

Login en el sistema por parte del alumno

Introducción: Existe la necesidad de saber quien accede al sistema para registrar la realización de las prácticas.

Entradas: Nombre de usuario, contraseña y práctica que se va a realizar.

Salida: Actualización de la base de datos con el registro del acceso a la sesión de prácticas indicada.

Consultar información personal

Introducción: Tenemos que conocer la información personal de cada usuario.

Entradas: Nombre de usuario del alumno o profesor.

Salida: Muestra la información referente al usuario como es el DNI, su nombre completo, la práctica realizada y la fecha del acceso al sistema.

Consultar prácticas realizadas por parte del alumno

Introducción: Debemos conocer las prácticas que hemos realizado y cuales han sido entregadas

Entradas: Nombre de usuario del alumno.

Salida: Muestra la información referente a las prácticas que ha realizado y entregado el alumno.

Consultar histórico de sesiones

Introducción: Necesitamos saber todas la veces que un usuario ha accedido al sistema.

Entradas: Nombre de usuario del alumno o profesor.

Salida: Muestra el registro de todas las veces que un usuario se ha conectado al sistema, mostrando la fecha, la hora y la práctica seleccionada.

Logout del sistema por parte del alumno

Introducción: Necesitamos cerrar la sesión del usuario.

Entradas: Nombre de usuario del alumno.

Salida: Actualiza el registro de conexión en la base de datos y cierra la sesión de usuario impidiendo el acceso a la información.

Login en el sistema por parte del profesor

Introducción: Existe la necesidad de saber quien accede al sistema para registrar la realización de las prácticas.

Entradas: Nombre de usuario, contraseña y práctica que se va a realizar.

Salida: Actualización de la base de datos con el registro del acceso a la sesión de prácticas indicada.

Consultar el mapa de la clase

Introducción: Existe la necesidad de saber qué alumnos están realizando las practicas actualmente.

Entradas: Código del aula.

Salida: Muestra la distribución del aula así como los alumnos que están realizando la práctica y en qué puestos de trabajo la está realizando.

Consultar las prácticas que ha realizado cada alumno

Introducción: El profesor necesita saber qué prácticas ha realizado cada alumno.

Entradas: Nombre de usuario del alumno.

Salida: Muestra las prácticas que ha realizado cada alumno y cuales han sido entregadas.

Marcar las prácticas de los alumnos como entregadas

Introducción: El profesor debe tener la posibilidad de marcar una práctica de un alumno como entregada.

Entradas: Nombre de usuario del alumno y la práctica.

Salida: Actualiza el registro de la base de datos correspondiente a la práctica del alumno y la guarda como práctica entregada. La muestra al profesor y al alumno como tal.

Logout de un alumno, de un grupo o de todos los alumnos

Introducción: Necesitamos cerrar la sesión de un usuario concreto o de un grupo de usuarios accediendo desde el mapa del aula.

Entradas: Nombre de usuario del alumno.

Salida: Actualiza el registro de conexión en la base de datos y cierra la sesión de los usuarios seleccionados.

Logout del sistema por parte del profesor

Introducción: Necesitamos cerrar la sesión del usuario y así conocer cuando se ha finalizado la sesión de prácticas.

Entradas: Nombre de usuario del alumno.

Salida: Actualiza el registro de conexión en la base de datos y cierra la sesión de profesor y de todos los alumnos que no habían sido cerradas.

2.3.2 Requisitos de interfaces externas

Interfaces de usuario

La aplicación se visualizará en una pantalla mediante el navegador web, dónde el usuario dispondrá de un menú para acceder a todos los servicios.

Mediante el teclado y el ratón podrá interactuar con el sistema e introducir los datos que el sistema le solicite.

Interfaces hardware

La aplicación puede utilizarse en todos los dispositivos que dispongan de navegador web.

No se necesitará ningún hardware especial y el dispositivo no tendrá ningún requisito mínimo a nivel de hardware para poder acceder a la aplicación.

Interfaces software

La aplicación puede utilizarse en cualquier sistema operativo mediante un navegador web existente en todos los sistemas operativos o en los dispositivos con acceso a Internet.

Interfaces de comunicación

La aplicación está diseñada para que se utilice a través de la red, la única restricción es que el dispositivo desde que se accede a la aplicación debe estar configurado por la aplicación como perteneciente al aula a la que se quiere acceder.

2.3.3 Requisitos de eficiencia

La aplicación esta desarrollada para cualquier sistema operativo y para cualquier dispositivo que disponga de navegador web capaz de interpretar archivos en PHP.

La aplicación se instalará en un servidor web que tenga las herramientas necesarias para interpretar el código fuente en PHP y que sea capaz de poder acceder a bases de datos en MySQL.

2.3.4 Restricciones de diseño

Estándares cumplidos

La aplicación debe cumplir las reglas de los navegadores web para que estos sean capaces de interpretar el código que les envía el servidor web y puedan mostrar correctamente todos los elementos que contiene la aplicación para que funcione correctamente.

Limitaciones hardware

Las limitaciones hardware de la aplicación son las limitaciones que posee el dispositivo desde el cuál se ejecute esta.

No existe ninguna restricción en cuanto a requisitos mínimos del sistema.

2.3.5 Atributos

Seguridad

Respetando los formatos de entrada del teclado y ratón en la aplicación, nos garantizaremos el buen funcionamiento de ella, también los errores en los dispositivos dónde se ejecuta la aplicación no deben afectar a esta porque se ejecuta en un servidor web.

Para acceder al sistema se necesita un usuario y contraseña, con ello se controla el acceso a la aplicación.

La aplicación contará con un sistema de seguridad que no permita el acceso al sistema a usuarios no registrados o sin permisos para utilizar la aplicación.

Un usuario que no esté realizando la práctica en el aula donde se desarrolla la sesión no podrá acceder al sistema. Por lo tanto un usuario no puede loguearse sino está en la aula correspondiente a su sesión de prácticas.

Mantenimiento

La aplicación no necesita un mantenimiento. Su instalación y actualización es muy rápida ya que simplemente tenemos que copiar su contenido en un servidor web y ejecutar las secuencias en la base de datos para crearla.

2.3.6 Otros requisitos

Bases de datos

Necesitaremos crear una base de datos para almacenar toda la información que le facilitemos a la aplicación.

En la base de datos se deberá almacenar los usuarios, ya sean alumnos o profesores que tienen permisos para acceder al sistema.

Se almacenarán los registros de entrada y salida a las sesiones de prácticas de cada alumno, así como las sesiones a las que ha asistido cada profesor.

Para almacenar todos los datos se utilizará una base de datos MySQL que estará en un servidor donde se ofrezca dicho servicio.

3. Análisis

Contiene la descripción de los conocimientos que deberá poseer cada usuario para poder utilizar la aplicación.

Contiene todos los escenarios posibles, describiendo los comportamientos de la aplicación en las situaciones más relevantes.

Se muestran los diagramas de secuencia en cada escenario.

3.1 Casos de uso

3.1.1 Actores

En esta sección se describirán las características de los usuarios que usarán la aplicación.

Existen dos tipos de actores que participan en la aplicación, los alumnos y los profesores.

Para la utilización del sistema, deberán estar dados de alta en la base de datos, loguearse en la aplicación como alumno o profesor y estar físicamente en el aula que les pertenece, en caso contrario, se mostrará un error y no podrán utilizarla.

Cualquier otro usuario no puede utilizar la aplicación.

En el diagrama 1, se muestra una vista general del entorno del sistema.

Se detalla graficamente los actores que intervienen en el sistema, alumno y profesor y se muestra el sistema que corresponde a la “gestión de las sesiones de prácticas”.



Diagrama 1

En las tablas siguientes detallamos las características que debe poseer cada actor:

Alumno registrado
Rol de usuario
Usuario
Experiencia tecnológica
Básico, que tenga conocimientos para navegar por Internet
Otras características
<ul style="list-style-type: none"> - Debe ser alumno de la universidad para poder utilizar el sistema - Debe de estar dado de alta en la base de datos del sistema - Debe de estar físicamente en el aula para poder loguearse

Profesor registrado
Rol de usuario
Administrador
Experiencia tecnológica
Experto, capaz de poder editar xml
Otras características
<ul style="list-style-type: none"> - Debe de estar físicamente en el aula para poder loguearse - Debe ser profesor de la universidad para poder utilizar el sistema - Debe tener un conocimiento de la aplicación para poder editar los archivos de la configuración - Debe tener permisos para cambiar archivos en el sistema operativo. - Debe de estar dado de alta en la base de datos del sistema

3.1.2 Escenarios

Los escenarios son secuencias de sucesos que se producen en el sistema.

En este apartado, detallaremos los escenarios que pertenecen a la aplicación, indicando el estado inicial del sistema, su flujo normal de eventos, las posibles situaciones de error y las posibles acciones a realizar.

Para cada escenario se definirá un diagrama de secuencias y una tabla describiendo el escenario.

La tabla consta de:

Nombre del escenario: Identifica el escenario.

Actores: Son los usuarios que aparecen en el escenario.

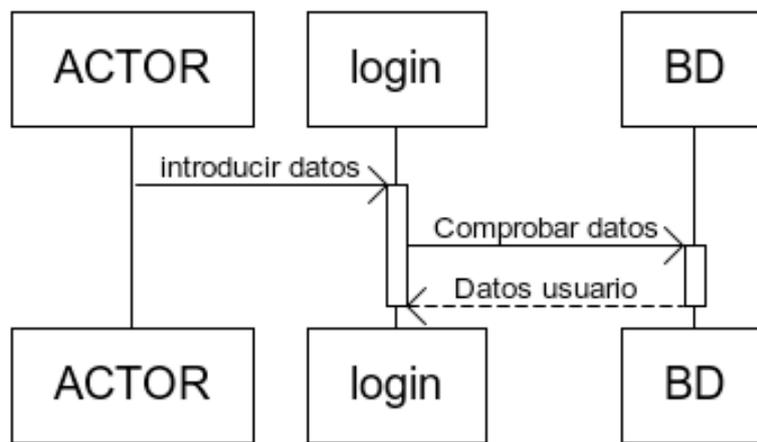
Descripción: Describe que función se realizará en el escenario.

Escenario principal: Describe los pasos que el actor va realizando.

Extensiones: Los posibles errores que puedan aparecer.

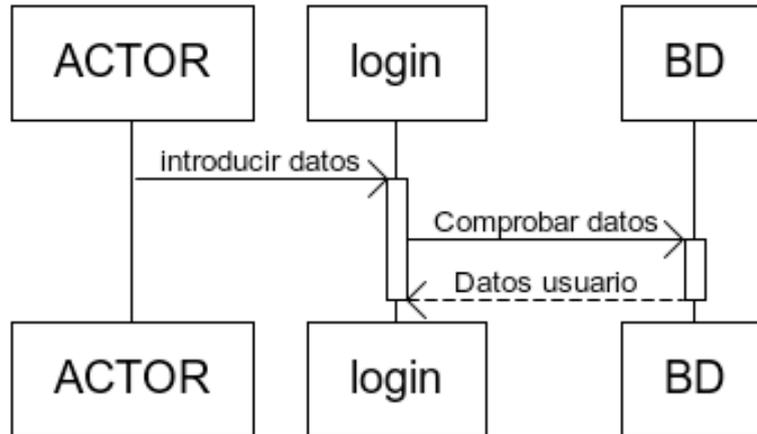
A continuación, mostramos todos los escenarios:

Registro de los alumnos a las sesiones de prácticas



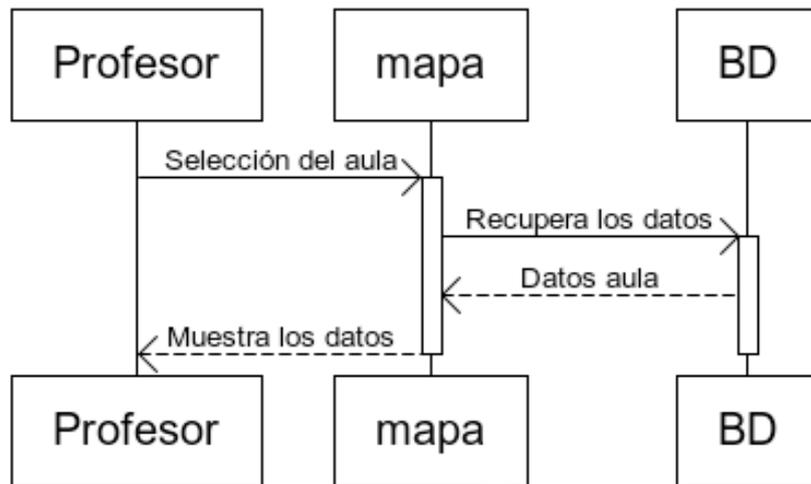
Registro de los alumnos a la sesión de prácticas
Actores
Alumno
Descripción
El sistema permite al alumno registrarse para empezar la sesión de prácticas
Escenario principal
<ol style="list-style-type: none"> 1. El alumno introduce sus datos de conexión en el sistema. 2. El sistema comprueba que los datos de conexión son correctos. 3. El sistema comprueba que el alumno pertenece a la sesión de prácticas a la que quiere acceder.
Extensiones
<p>2a. Los datos de conexión son erróneos</p> <p>2a1, El sistema informa al alumno que los datos son incorrectos y le vuelve a solicitar los datos, vuelve al paso 1.</p> <p>3a. El alumno no está accediendo desde un puesto de trabajo perteneciente a la sesión de prácticas, el sistema le informa al alumno de la situación actual.</p>

Registro de los profesores a las sesiones de prácticas



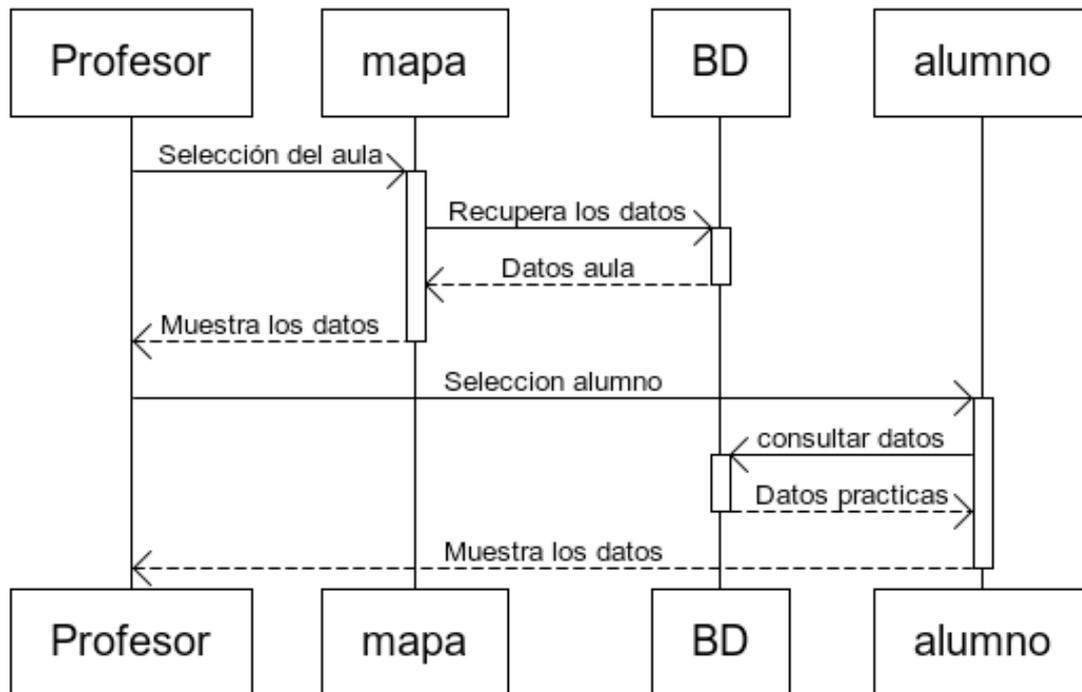
Registro del profesor a la sesión de prácticas
Actores
Profesor
Descripción
El sistema permite al profesor registrarse para empezar la sesión de prácticas
Escenario principal
<ol style="list-style-type: none">1. El profesor introduce sus datos de conexión en el sistema.2. El sistema comprueba que los datos de conexión son correctos.3. El sistema comprueba que el profesor está accediendo desde un puesto de trabajo como profesor y en el aula a la que pertenece la sesión de prácticas
Extensiones
<p>2a. Los datos de conexión son erróneos</p> <p>2a1, El sistema informa al profesor que los datos son incorrectos y le vuelve a solicitar los datos, vuelve al paso 1.</p> <p>3a. El profesor no está en un puesto de trabajo perteneciente a la sesión de prácticas y la aplicación le informa de ello.</p>

Consultar distribución del aula



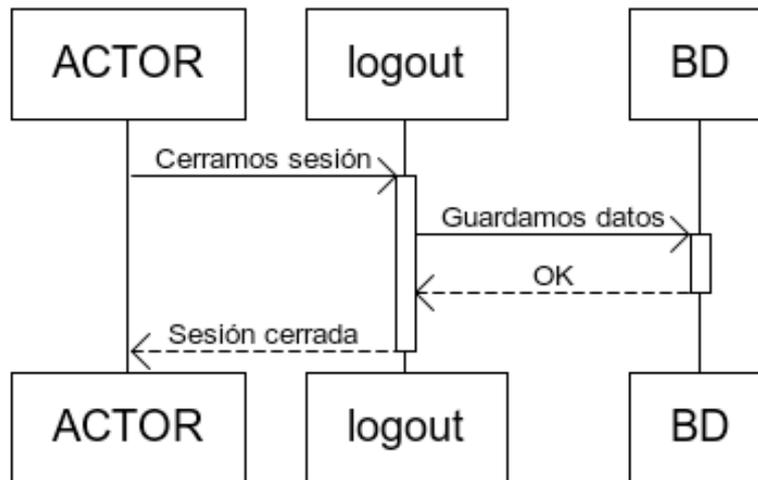
Consultar distribución aula
Actores
Profesor
Descripción
El sistema permite al profesor visualizar dónde están sentados todos los alumnos en clase y las prácticas que está realizando cada uno
Precondición
El profesor debe estar registrado en el sistema
Escenario principal
<ol style="list-style-type: none">1. El profesor indica al sistema que desea visualizar un mapa.2. El sistema muestra las aulas disponibles3. El profesor selecciona de entre todas las aulas, la aula de la cual quiere obtener su distribución4. El sistema recupera la información del aula y la muestra por pantalla.

Modificar el estado de una práctica



Modificar estado de una práctica
Actores
Profesor
Descripción
El sistema permite al profesor modificar el estado de una práctica
Precondición
El profesor y el alumno a modificar deben estar registrado en el sistema
Escenario principal
<ol style="list-style-type: none"> 1. El profesor indica al sistema que desea visualizar un mapa. 2. El sistema muestra las aulas disponibles 3. El profesor selecciona de entre todas las aulas, la aula de la cual quiere obtener su distribución 4. El sistema recupera la información del aula y la muestra por pantalla. 5. El profesor presiona sobre el nombre del alumno mostrado en la distribución del aula. 6. El sistema recupera el listado de las prácticas realizadas por el alumno seleccionado. 7. El profesor presiona sobre la práctica a la que desea cambiar su estado. 8. El sistema almacena el cambio en la base de datos.

Logout profesor



Logout profesor
Actores
Profesor
Descripción
El sistema permite al profesor cerrar su sesión y la de todos los alumnos conectados.
Precondición
El profesor debe estar registrado en el sistema
Escenario principal
1. El profesor indica al sistema que quiere cerrar la sesión. 2. El sistema cierra la sesión del profesor y de todos los alumnos conectados en ese momento.

4. Diseño

Contiene una introducción sobre la arquitectura de software utilizada, describiendo la estructura del servidor en tres capas: Presentación, lógica de control o negocio y persistencia.

Se describe la estructura de la base de datos explicando todos los campos que la componen.

4.1 Arquitectura software

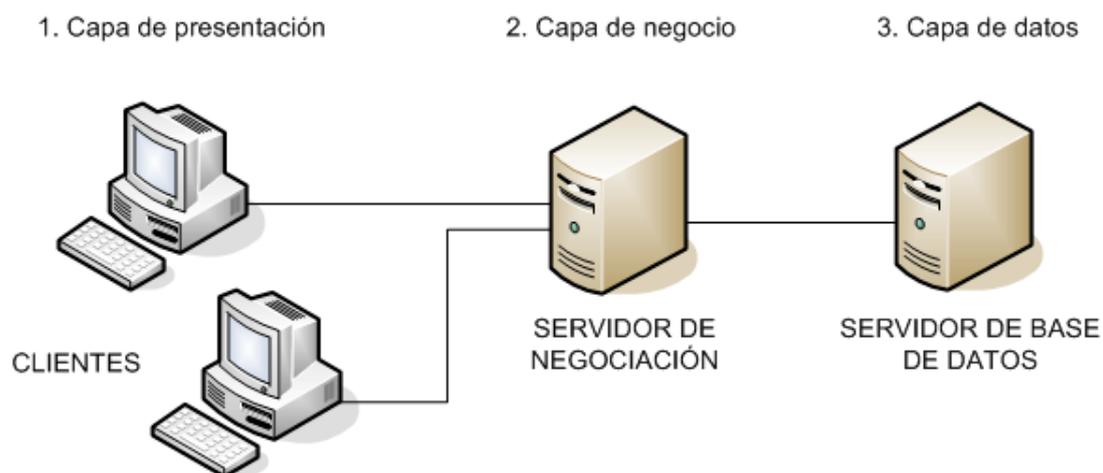
La arquitectura de software es el diseño de más alto nivel de la estructura de un sistema o aplicación dónde se definen sus módulos principales y que responsabilidades va a tener cada módulo, así como la interacción entre ellos.

Así podríamos definir a la arquitectura de software como la organización fundamental de un sistema formada por sus componentes, las relaciones entre ellos y el contexto en el que se implantarán, y los principios que orientan su diseño y evolución.

Una de las arquitectura de software más utilizada es la de arquitectura separada por 3 capas, que consiste en separar por partes la programación de una aplicación, es decir, separar la parte visible de la parte de almacenamiento o persistencia de datos.

Una de las principales ventajas de este tipo de desarrollo es que si hay que realizar un cambio sólo se debe realizar en la capa a la que afecta sin necesidad de revisar el resto del código de las otras capas.

Otra ventaja es la distribución de trabajo por equipos, ya que unas personas pueden estar desarrollando la capa de presentación, mientras otro equipo puede estar desarrollando la capa de negocio.



Como hemos comentado anteriormente, hemos decidido utilizar una arquitectura de 3 capas en el que nuestro proyecto está alojado en un servidor web y dispone de un servicio de bases de datos phpmyadmin.

4.1.1 Capa de presentación

Es la capa visible por parte del usuario, encarga de presentar y capturar la información por parte del usuario. Esta capa se comunica únicamente con la capa de negocio o lógica de control.

En nuestra aplicación, la capa de presentación consiste en la visualización de una página web, encargada de mostrar toda la información que el usuario solicita así como enviar y almacenar los datos que le son introducidos.

La aplicación interactúa con el usuario, por lo que es necesario que ésta le muestre información y deba obtener información de la persona que la utiliza.

Para obtener la información del usuario, la aplicación cuenta con las herramientas web que ofrecen los estándares de los navegadores.

En este ejemplo, existen campos de texto y desplegables dónde la aplicación puede obtener información por parte del usuario para almacenarla en la base de datos.



REGISTRO ASISTENCIA A PRÁCTICAS

Para iniciar la sesión de prácticas es necesario logearse como usuario de la UPV.
En caso de realizar la práctica en pareja, se deberá incluir el usuario y clave de los alumnos que realizan la práctica.

	Alumno 1		Alumno 2
Usuario	<input type="text"/>	Usuario	<input type="text"/>
Clave	<input type="text"/>	Clave	<input type="text"/>
<input type="button" value="Practica 1"/>			
<input type="button" value="Enviar consulta"/>			

También es capaz de mostrar información que la aplicación tiene almacenada en la base de datos, como vemos en el siguiente ejemplo:



Histórico

<i>Fecha</i>	<i>ENTRADA Hora</i>	<i>SALIDA Hora</i>	<i>Práctica</i>
Mie, 09 Feb 2011	16:02:51	16:03:52	Practica 1
Lun, 14 Feb 2011	10:56:51	11:02:06	Practica 1
Lun, 14 Feb 2011	11:02:14	11:02:18	Practica 1
Lun, 14 Feb 2011	11:02:26	11:19:08	Practica 2
Lun, 14 Feb 2011	11:19:18	11:21:15	Practica 3
Sab, 16 Jul 2011	12:14:27		Practica 1

En este caso, la aplicación recupera de la base de datos la información referente a un alumno, una vez ha obtenido los datos oportunos, la capa de negocio que veremos en el apartado siguiente, se encarga de procesar los datos y en la capa de presentación se muestran en el formato correcto para que sea fácil de comprender y de interpretar por parte del usuario.

4.1.2 Capa de negocio o lógica de control

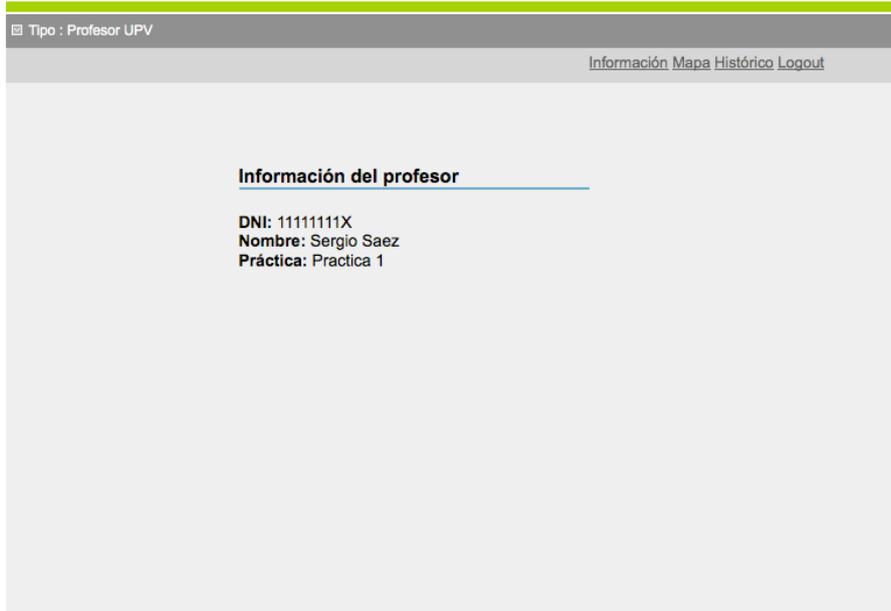
La capa de negocio es donde reside la aplicación y se ejecuta. Recibe las peticiones del usuario y se envían las respuestas tras el proceso.

Esta capa se comunica con la capa de presentación para poder recibir las solicitudes por parte del usuario y presentar los resultados.

También se comunica con la capa de datos para solicitar al gestor de la base de datos para almacenar y recuperar los datos de la base de datos.

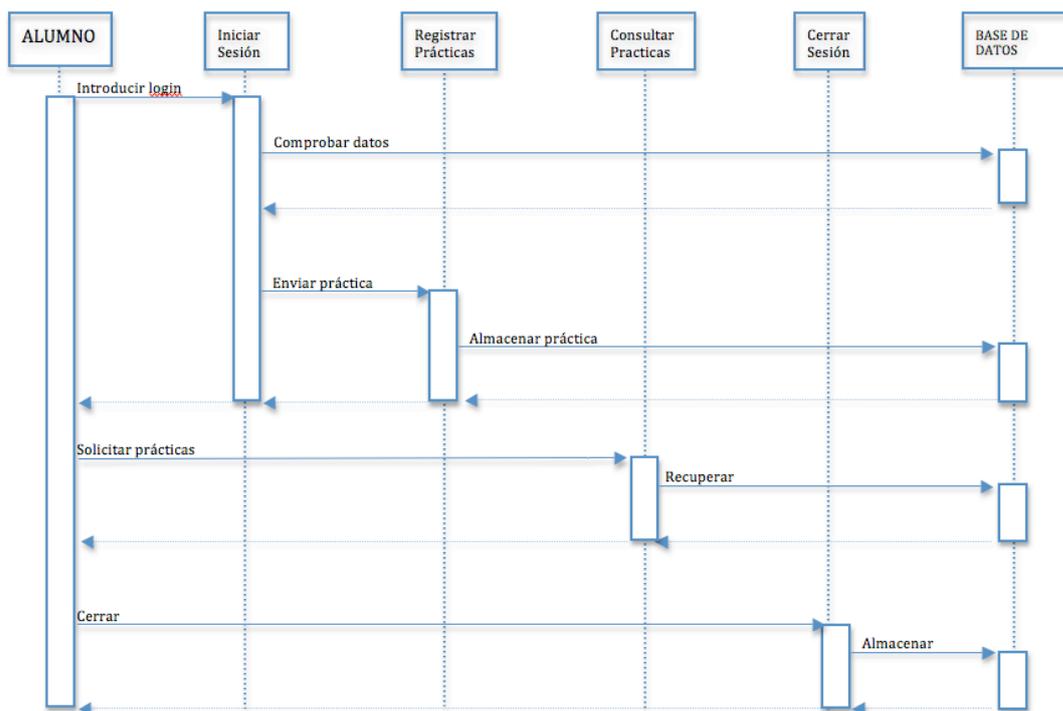
El servidor dónde está alojado nuestro proyecto se encarga de atender las peticiones del usuario y nuestra aplicación ejecuta las funciones necesarias para mostrarle al usuario la información que necesita o realizar las llamadas para que se realicen las consultas a la base de datos en la capa de persistencia.

Para visualizar la información referente a un profesor, en la capa de negocio o lógica de control se ha recuperado los datos del profesor de la base de datos comprobando que los datos que necesitaba para obtener la información eran correctos.



Una vez recuperada la información que ha obtenido de la base de datos, los ha tratado para que no haya errores y en la capa de presentación han sido formateados para que se muestren correctamente.

Con el siguiente diagrama de secuencia podemos observar las funciones que podemos realizar con la aplicación.



4.1.3 Capa de persistencia

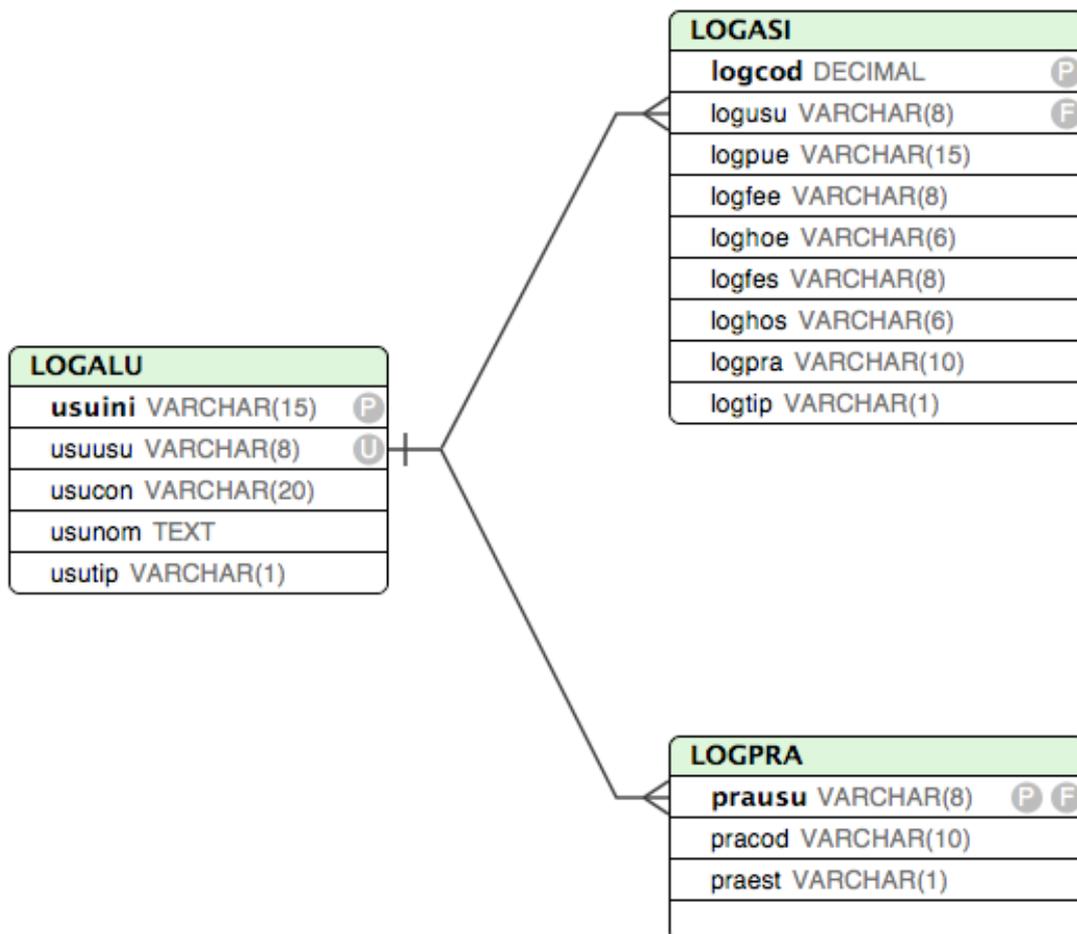
La capa de persistencia es dónde residen los datos y se encarga de acceder a

ellos. Está formado por un gestor de bases de datos que realiza la lectura y almacenamiento de datos.

En nuestro proyecto, es una base de datos phpmyadmin alojada en nuestro servidor. Contiene toda la información que el usuario transfiere a la aplicación o que la propia aplicación va almacenando según las acciones del usuario.

Consiste en 3 tablas:

- LOGALU: Contiene la información del usuario, como sus datos personales y tipo de usuario.
- LOGASI: Se almacenan los datos referentes a la asistencia a las prácticas por parte del usuario, cuando acude a las prácticas, cuando abandona la sesión y qué prácticas ha realizado.
- LOGPRA: Contiene el estado de las prácticas para cada usuario.



LOGALU

usuini: Dni del usuario

usuusu: Nombre de usuario

usucon: Contraseña

usunom: Nombre real del usuario

usutip: Tipo de usuario (A: Alumno o P: Profesor)

LOGASI

logcod: Código numérico

logusu: Nombre de usuario

logpue: Ip del puesto de trabajo

logfee: Fecha de entrada

loghoe: Hora de entrada

logfes: Fecha de salida

loghos: Hora de salida

logpra: Práctica que ha realizado

LOGPRA

prausu: Nombre de usuario

pracod: Código de la práctica

praest: Estado de la práctica

5. Implementación

En este apartado describimos cómo ha sido programada la aplicación y la tecnología utilizada, así como la estructura que contiene y la explicación del contenido de cada fichero que lo forma.

5.1 Tecnología

Uno de los requisitos para la realización del proyecto es que la aplicación debía desarrollarse en PHP, por lo que la tecnología elegida para desarrollar el proyecto es PHP.

Todo el código del proyecto se ha realizado con PHP utilizando las funciones de la versión 5.3, para la realización de la captura de datos, su tratamiento y almacenamiento.

Para el almacenamiento de los datos se ha utilizado phpmyadmin que se integra perfectamente con PHP.

La programación de la parte visual del proyecto, es decir, la visualización del proyecto se ha utilizado HTML que es el lenguaje estándar para mostrar contenidos en las páginas web.

Los estilos de la página web se han realizado con CSS, el cual se combina con el HTML para lograr una correcta visualización de la aplicación y que esta sea accesible por parte del usuario.

También se ha utilizado Javascript para realizar funciones específicas, como la ocultación de divs al presionar un botón o mostrarlo cuando se vuelve a presionar.

La programación del código del proyecto se ha utilizado la aplicación “Eclipse” en su versión Galileo que incorpora las librerías de PHP y Javascript.

Para la realización de la estructura de la base de datos se ha utilizada “SQL Editor”, aplicación que funciona en Mac Osx para generar y construir diagramas y bases de datos.

El sistema dónde se ha programado la totalidad del proyecto ha sido en un Macbook con Snow Leopard.

El proyecto está alojado en un servidor de la compañía OVH que nos ofrece el servicio de alojamiento y la base de datos.

5.2 Metodología

No se utiliza ningún framework para la realización del proyecto por lo que todo se ha realizado partiendo desde cero. Antes de la realización de la aplicación se ha dedicado un tiempo para el estudio de la distribución de los ficheros del proyecto así como la nomenclatura a utilizar.

5.3 Estructura de la aplicación

En la aplicación existen 3 directorios principales: Ofun, Oima, 1htm.

La elección de los nombres es para que haya una ordenación en el directorio, ya que 0fun contiene las funciones que se utilizan en la aplicación, 0ima contiene las imágenes y 1htm contiene todo los archivos PHP de la aplicación.

- /0fun
 - o funbda01.php
Contiene todas las funciones que utiliza la aplicación que acceden a la base de datos
 - o funhtm.php
Contiene todas las funciones que muestras componentes HTML
 - o funjvs01.js
Contiene todas las funciones javascript que utiliza la aplicación.

- /0ima
Contiene todas las imágenes que muestra la aplicación.

- /1htm
 - o /xml: Contiene archivos xml
 - aulas.xml
Contiene la distribución del aula
 - practicas.xml
Contiene las prácticas del aula

 - o ctlhtm.php: Controlador de los archivos
 - o htma1.php: Login alumno
 - o htma2.php: Validación alumno
 - o htma4.php: Información alumno
 - o htma5.php: Información prácticas alumno
 - o htma6.php: Información histórico alumno
 - o htma7.php: Logout alumno

 - o htmb1.php: Login profesor
 - o htmb2.php: Validación profesor
 - o htmb3.php: Información profesor
 - o htmb5.php: Formulario del aula
 - o htmb6.php: Distribución de las aulas
 - o htmb7.php: Histórico profesor
 - o htmb8.php: Logout profesor
 - o htmb9.php: Estado prácticas alumnos
 - o htmb10.php: Cambiar estado prácticas alumnos
 - o htmb11.php: Logout alumnos

- index.php: Página inicial
- gensty01.css: Hoja estilos para el html

6. Pruebas de uso

Describimos las pruebas que se han realizado para el buen comportamiento de la aplicación.

Mostramos los resultados obtenidos y que hemos aplicado para realizar la mejora de resultados.

6.1 Pruebas realizadas

A medida que se iba desarrollando la aplicación se fueron realizando pruebas con los requisitos y especificación detalladas anteriormente. Se han desarrollado diferentes tipos de pruebas:

6.1.1 Pruebas funcionales

Las pruebas funcionales han consistido en la comprobación de cada una de las funciones que realiza la aplicación. Una vez programadas se mostraba por pantalla el resultado de la función, por lo que cada función ha sido comprobada unitariamente.

Se ha sometido a la aplicación a la introducción de datos erróneos para comprobar como responde así como actúa frente a este tipo de errores. Uno de los pocos casos en los que se le pueden pasar datos erróneos a la aplicación es en el sistema de login, ya que en los demás apartados, se selecciona la información mediante una lista desplegable por lo que es imposible que se le pueda introducir datos erróneos a la aplicación.

Para el caso de login, se ha diseñado la función para que no permita la introducción de “inyecciones SQL”. Estas inyecciones consisten en la introducción de un código intruso que se vale de los errores en la programación a nivel de validación para que el texto introducido genere una consulta maliciosa en la base de datos.

Una de las inyecciones más utilizadas es introducir en el nombre de usuario cualquier texto y en la contraseña el siguiente texto ' OR 'x' = 'x con ello intentando “engañar” a la base de datos.

Hemos introducido las inyecciones SQL más comunes y el resultado de las mismas es el siguiente:



Tipo : Alumno UPV

*ERROR, el usuario asdfags no se puede logear porque el usuario o la contraseña es incorrecta.

REGISTRO ASISTENCIA A PRÁCTICAS

Para iniciar la sesión de prácticas es necesario logearse como usuario de la UPV.
En caso de realizar la práctica en pareja, se deberá incluir el usuario y clave de los alumnos que realizan la práctica.

	Alumno 1		Alumno 2
Usuario	<input type="text"/>	Usuario	<input type="text"/>
Clave	<input type="text"/>	Clave	<input type="text"/>
<input type="button" value="Practica 1"/>			
<input type="button" value="Enviar"/>			

6.1.2 Pruebas de visualización

Las pruebas de visualización han sido las encargas de comprobar que la aplicación se visualizaba correctamente en los navegadores que existen en el mercado y en la mayoría de los dispositivos capaces de mostrar una aplicación web.

Firefox



Tipo : Alumno UPV

REGISTRO ASISTENCIA A PRÁCTICAS

Para iniciar la sesión de prácticas es necesario logearse como usuario de la UPV.
En caso de realizar la práctica en pareja, se deberá incluir el usuario y clave de los alumnos que realizan la práctica.

	Alumno 1		Alumno 2
Usuario	<input type="text"/>	Usuario	<input type="text"/>
Clave	<input type="text"/>	Clave	<input type="text"/>
<input type="button" value="Practica 1"/>			
<input type="button" value="Enviar consulta"/>			

Safari



UNIVERSIDAD
POLITECNICA
DE VALENCIA

Escuela Técnica Superior de Ingeniería Informática



Tipo : Alumno UPV

REGISTRO ASISTENCIA A PRÁCTICAS

Para iniciar la sesión de prácticas es necesario logearse como usuario de la UPV.
En caso de realizar la práctica en pareja, se deberá incluir el usuario y clave de los alumnos que realizan la práctica.

	Alumno 1		Alumno 2
Usuario	<input type="text"/>	Usuario	<input type="text"/>
Clave	<input type="text"/>	Clave	<input type="text"/>
<input type="button" value="Practica 1"/>			
<input type="button" value="Enviar"/>			

Iphone

6.1.3 Pruebas de tiempo de ejecución

Las pruebas de tiempo de ejecución han consistido en comprobar si el tiempo de cálculo de las funciones no ha excedido de lo habitual y no retrasa el funcionamiento o la ejecución de toda la aplicación.

Para realizar la medición del tiempo de ejecución en las funciones de PHP se han utilizado herramientas que ya dispone el propio lenguaje de programación, estas funciones consisten en la introducción de un código fuente en las funciones más relevantes de la aplicación o en las funciones que podrían retrasar el tiempo de ejecución comprobando el tiempo que tardan en finalizar.

Para ello, al comienzo de la función introducimos una función de PHP que contiene la hora UNIX en una variable y al finalizar nuestra función restamos la hora final con la hora inicial y con esta operación podemos calcular el tiempo de ejecución de la función.

No ponemos los tiempos de las funciones ya que son despreciables, el tiempo de ejecución de las funciones es mínimo aunque en el apartado siguiente podremos observar el tiempo de carga de la aplicación.

6.1.4 Pruebas de duplicidad de código

En las pruebas de duplicidad de código se comprueba que no hay código duplicado y en caso de encontrar líneas de código que realizan las mismas funciones, se ha factorizado para conseguir la optimización del código, eliminando código repetido y mejorando su compresión y ejecución.

6.1.5 Pruebas de uso

Se han realizado pruebas de uso de la aplicación, con ello conseguimos mejorarla para que sea accesible, con una rápida respuesta de las acciones a realizar, eficiente y que el usuario comprenda correctamente el contenido de la misma.

Estas pruebas se han basado en 4 aspectos:

Accesibilidad:

La aplicación es de fácil acceso, los controles son simples y es muy fácil acceder a cualquier contenido de la misma. A continuación, podemos observar unas capturas de la aplicación para comprobar su simpleza y optimización de contenidos.

Login



UNIVERSIDAD
POLITECNICA
DE VALENCIA

Escuela Técnica Superior de Ingeniería Informática



Tipo : Profesor UPV

REGISTRO ASISTENCIA A PRÁCTICAS

Desde aquí el profesor podrá controlar la asistencia a clases de sus alumnos y saber dónde están sentados.

Usuario

Clave

Practica 1

Este apartado es el login del usuario, cada usuario deberá tener un nombre de usuario y una clave.

Para acceder a la aplicación basta con introducir su usuario y clave y presionar en “Enviar”.

Si ha introducido correctamente los datos, accederá al sistema, en caso contrario, se le mostrará un mensaje de error informándole de lo que ha ocurrido.

Información de usuario



UNIVERSIDAD
POLITECNICA
DE VALENCIA

Escuela Técnica Superior de Ingeniería Informática



Tipo : Profesor UPV

[Información](#) [Mapa](#) [Histórico](#) [Logout](#)

Información del profesor

DNI: 11111111X
Nombre: Sergio Saez
Práctica: Practica 1

Al acceder a la aplicación se le mostrará la información correspondiente a su perfil, en el caso de la captura se muestra la información del profesor pero para el alumno es exactamente igual.

Contiene el DNI del usuario, su nombre y la práctica que está realizando en el caso del alumno y si el usuario que está logueado es un profesor se mostrará la práctica que ha elegido en el login como la práctica que corresponde a la sesión que se está realizando.

Historial



<i>Fecha</i>	<i>ENTRADA Hora</i>	<i>SALIDA Hora</i>	<i>Práctica</i>
Mie, 09 Feb 2011	16:02:51	16:03:52	Practica 1
Lun, 14 Feb 2011	10:56:51	11:02:06	Practica 1
Lun, 14 Feb 2011	11:02:14	11:02:18	Practica 1
Lun, 14 Feb 2011	11:02:26	11:19:08	Practica 2
Lun, 14 Feb 2011	11:19:18	11:21:15	Practica 3
Sab, 16 Jul 2011	12:14:27		Practica 1

Con el historial podremos observar nuestra actividad, las prácticas a las que hemos ido y cuando hemos entrado y salido de las aulas.

Nos permite tener un control de nuestra asistencia a las prácticas y conocer rápidamente que prácticas nos faltan por realizar.

Cuando la fecha de salida no sea rellenada significa que la práctica la estamos haciendo en ese momento en caso contrario, la fecha de salida se almacena una vez cerramos la sesión de prácticas.

Distribución de las aulas

UNIVERSIDAD POLITÉCNICA DE VALENCIA Escuela Técnica Superior de Ingeniería Informática etsinf

Tipo : Profesor UPV

[Información](#) [Mapa](#) [Histórico](#) [Logout](#)

 158.42.181.22	 158.42.181.21	 158.42.181.20	 158.42.181.19	 158.42.181.18
 158.42.181.17	 158.42.181.16	 158.42.181.15	 158.42.181.14	 158.42.181.13
 158.42.181.12	 158.42.181.11	 158.42.181.10	 158.42.181.9	 158.42.181.8
 158.42.181.7	 158.42.181.6	 158.42.181.5	 158.42.181.4	 158.42.181.3
		 158.42.181.2		 158.42.181.1
				

Logout alumnos

Universidad Politécnica de Valencia

Este apartado solamente es visible por el profesor, ya que puedes ver todos lo alumnos conectados en cada ordenador y la práctica que está realizando.

Por último, en la pestaña de “Logout” simplemente al acceder a ella, se cierra la sesión de la aplicación y se da por finalizada la sesión de prácticas.

Rapidez de respuesta

El contenido de la aplicación es simple y la carga de las imágenes es muy liviana por lo que al realizar las llamadas a una página específica de la aplicación el tiempo de respuesta es mínimo así como la carga y visualización de su contenido.

Se han optimizado las imágenes para una correcta y rápida visualización, ya que su peso es mínimo y facilita la carga rápida.

Para comprobar el tiempo de carga, hemos recurrido a las utilidades que nos ofrece el navegador Google Chrome, en su apartado de Network podemos observar el tiempo de carga de una página web.

Antes de realizar este proceso, hemos eliminado la cache del navegador, ya que la aplicación se queda cacheada al estar usándose habitualmente y esto hace que los tiempos aún sean más reducidos.

En nuestro caso, para la versión en cache, el tiempo total de carga de la web es de 304 milisegundos:

Name Path	Method	Status Text	Type	Size Content	Time Latency	Timeline
/pfc/	GET	200 OK	text/html	1.57KB 3.66KB	237ms 235ms	
gensty01.css /pfc/	GET	(from cac...)	text/css	(from cac...)	0ms 0ms	
imacab01.gif /pfc/0ima	GET	(from cac...)	image/gif	(from cac...)	0ms 0ms	
imacab02.gif /pfc/0ima	GET	(from cac...)	image/gif	(from cac...)	0ms 0ms	
imaidi01.jpg /pfc/0ima	GET	(from cac...)	image/jpeg	(from cac...)	Pending	
imapie01.jpg /pfc/0ima	GET	(from cac...)	image/jpeg	(from cac...)	Pending	

6 requests | 1.57KB transferred | 304ms (onload: 304ms, DOMContentLoaded: 305ms)

Para la ejecución por primera vez de la aplicación, el tiempo se incrementa a 1,05 segundos:

Name Path	Method	Status Text	Type	Size Content	Time Latency	Timeline
/pfc/	GET	200 OK	text/html	1.57KB 3.66KB	458ms 444ms	
gensty01.css /pfc/	GET	200 OK	text/css	4.01KB 3.65KB	222ms 119ms	
imacab01.gif /pfc/0ima	GET	200 OK	image/gif	5.30KB 5.02KB	221ms 130ms	
imacab02.gif /pfc/0ima	GET	200 OK	image/gif	4.54KB 4.18KB	262ms 145ms	
imaidi01.jpg /pfc/0ima	GET	200 OK	image/jpeg	11.32KB 11.04KB	301ms 124ms	
imapie01.jpg /pfc/0ima	GET	200 OK	image/jpeg	11.33KB 11.04KB	370ms 195ms	

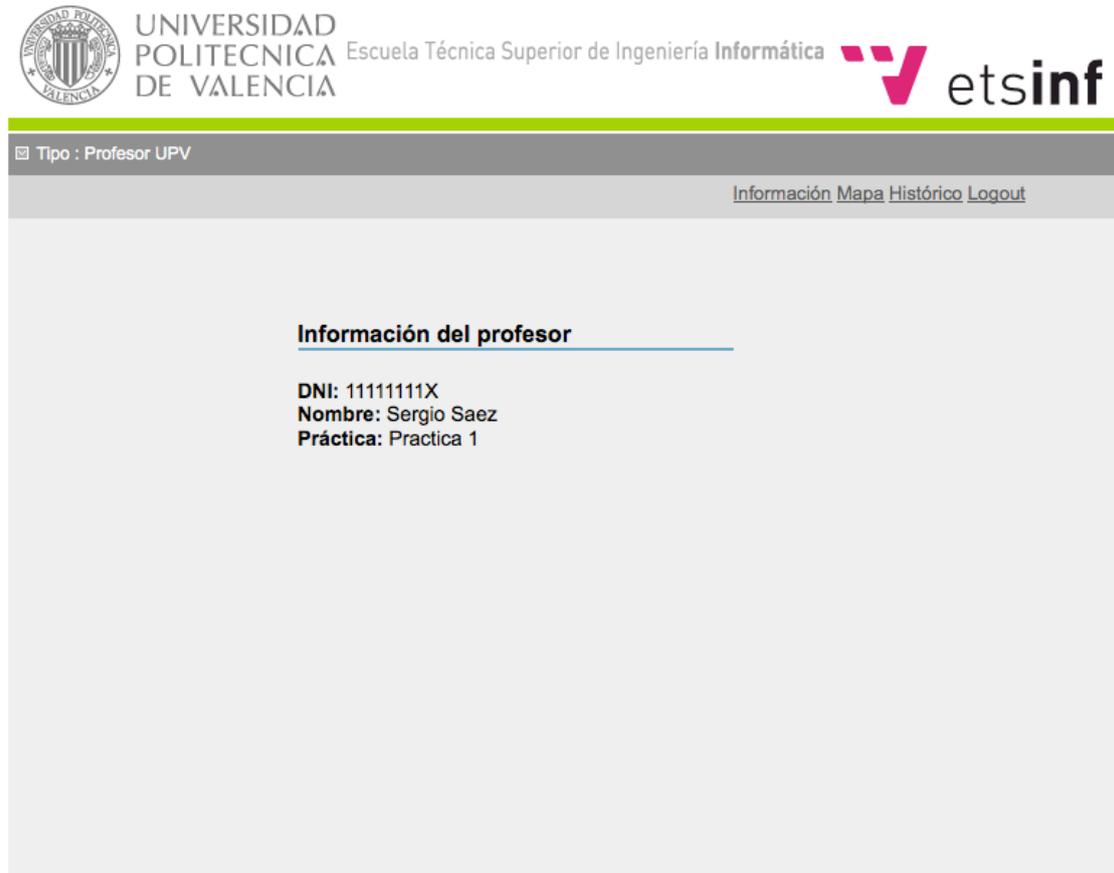
6 requests | 38.08KB transferred | 1.05s (onload: 1.05s, DOMContentLoaded: 680ms)

Si estudiamos con detalle los tiempos de carga, llegamos a la conclusión de que estos tiempos no son relevantes para la carga y ejecución de la aplicación ya que la página es muy liviana y esto hace que el tiempo de carga sea mínimo.

Eficiencia

Consiste en realizar los mínimos pasos para realizar una acción. Se ha estudiado todas las acciones que se deben realizar en la aplicación y se ha optimizado al máximo para que el usuario pueda realizar todas las acciones con los mínimos clicks posibles.

Como se puede observar en la imagen, se ha añadido un menú en la parte superior dónde puede consultar todas las acciones de la aplicación.



The screenshot displays the user interface of the etsinf application. At the top left, there is the logo of the Universidad Politécnica de Valencia and the text 'UNIVERSIDAD POLITECNICA DE VALENCIA'. To the right, it says 'Escuela Técnica Superior de Ingeniería Informática' and the 'etsinf' logo. Below this is a navigation bar with a dropdown menu showing 'Tipo : Profesor UPV' and links for 'Información', 'Mapa', 'Histórico', and 'Logout'. The main content area is titled 'Información del profesor' and displays the following details: 'DNI: 11111111X', 'Nombre: Sergio Saez', and 'Práctica: Practica 1'.

Comprensión

Las funcionalidades a realizar por parte del usuario han sido optimizadas para que sea de fácil comprensión para el usuario.

Se ha realizado un estudio del vocabulario mostrado en ella para que el usuario comprenda correctamente qué es lo que va a realizar cuando presione sobre un botón o acción.

El vocabulario de la aplicación no es técnico para su fácil comprensión.

7. Conclusiones

Descripción de los objetivos cumplidos con la realización de proyecto, conclusiones finales y de las posibles ampliaciones del mismo.

7.1 Objetivos cumplidos

A la terminación del proyecto, se puede analizar los objetivos planteados inicialmente y comprobar si se han cumplido.

Una vez se implemente la aplicación se podrá comprobar que realizar la evaluación de la asistencia a las sesiones de prácticas es más eficaz y eficiente que nombrar uno a uno cada alumno.

Ahora se puede conocer qué práctica está realizando cada alumno y en cada puesto de trabajo por lo que facilita la tarea al responsable de la sesión de prácticas.

También se puede comprobar qué prácticas han sido entregadas y si un alumno se presentó a la sesión con anterioridad.

7.2 Ampliaciones

Llegados a este punto, la aplicación permite su posible ampliación, ya que está realizada para una sola aula de prácticas pero esta podría permitir la implementación de varias aulas y en varios horarios.

También se podría conectar la aplicación con la base de datos de alumnos de la Universidad Politécnica de Valencia para mejorar el sistema de login y así unificar registros de usuarios, con un solo usuario y contraseña se pueden acceder a todos los servicios, ya que actualmente la base de datos está implementada en un Phpmyadmin.

7.3 Conclusión personal

Realizar el proyecto final de carrera y ponerlo en funcionamiento me ha servido para conocer y practicar la programación PHP y MySQL

La realización de la memoria del proyecto y explicación de todo lo que se ha realizado me sirve para enfrentarme al problema de describir todo lo que he desarrollado y lo que no se ha podido desarrollar.

Esta tarea ha sido la más costosa y complicada para mí, ya que no estoy acostumbrado a redactar y detallar los pasos seguidos para la realización de un proyecto, por ello ha sido más fácil desarrollar la aplicación que redactar toda la memoria.

He intentado realizar una memoria clara y amena para que el lector comprenda cómo se ha realizado el proyecto.

Exponiendo todo lo que se ha realizado e intentando plasmar con capturas de pantalla de la aplicación.

La conclusión final es que la realización del proyecto me ha aportado cosas muy positivas así como la experiencia de enfrentarte a los problemas de desarrollar una aplicación, teniendo que crear la base de datos y todas las funciones para cumplir los objetivos de la aplicación.

He mejorado mi nivel de programación en PHP así como la realización de consultas a una base de datos MySQL.

También he puesto en práctica mis conocimientos en HTML y CSS para que las páginas se muestren correctamente en todos los navegadores, incluyendo los dispositivos móviles como teléfonos o tablets.

8. Bibliografía

Detalle de todos los libros o sitios web dónde se ha obtenido la información para poder realizar el proyecto o que han servido de ayuda para poder desarrollarlo.

8.1 Libros

- Sommerville, I. Ingeniería del Software. (6a ed.).
Addison-Wesley, 2002
- Desarrollo Web con PHP 6, Apache y Mysql.
Anaya Multimedia, 2009
- CSS, Hoja de estilo en cascada para diseño Web
Anaya Multimedia, 2006
- CSS Práctico
Anaya Multimedia, 2006
- IEEE Recommended Practice for Software Requirements Specifications
- Apuntes asignatura “Ingeniería del Software de Gestión”

8.2 Sitios Web de referencia

- Manual PHP
<http://es.php.net/>
- Wikipedia
<http://es.wikipedia.org>
- Wamp Server
<http://www.wampserver.com/en/>

9. Anexo A

Contiene un manual para la instalación y configuración de la aplicación, así como un manual de uso.

9.1 Instalación

Existen varias posibilidades para la instalación de la aplicación dependiendo de dónde se quiera instalar.

La primera opción es contratar un servicio de alojamiento en Internet que admita la tecnología PHP y la posibilidad de creación de bases de datos en MySQL.

Sólo deberemos subir el contenido de las carpetas de la aplicación en la raíz de las carpetas del servidor.

También deberemos crear la base de datos, para ello nos dirigimos al panel de control que nos ofrecerá nuestro proveedor de alojamiento dónde podremos configurar las bases de datos que necesitamos.

En la raíz de la aplicación existe un archivo que se llama “localhost.sql” desde el cuál se puede crear la base de datos y todas sus tablas con los contenidos necesarios para la ejecución de la aplicación. Deberemos ir a la sección de SQL dentro del phpmyadmin que habremos accedido desde el panel de control de nuestro proveedor de alojamiento. Ahí deberemos ejecutar las sentencias SQL para la creación de la base de datos y todas sus tablas.

La segunda opción para la instalación es realizarla en un ordenador desde el cuál se pueda utilizar como un servidor de Internet, para ello deberemos instalar un programa que haga actuar a este ordenador como un servidor de alojamiento de Internet, en caso de ser un ordenador con Windows recomendaremos la instalación de Wamp, que lleva todos los paquetes necesarios para la ejecución de nuestra aplicación. En caso de ser Linux recomendamos una versión para Linux llamada Lampp.

Una vez instalada y ejecutada la aplicación Wamp o Lampp, deberemos copiar en la carpeta “www” nuestro proyecto.

Para la creación de la base de datos deberemos escribir en nuestro navegador web la dirección <http://localhost/phpmyadmin/> con la que podremos acceder a la gestión de las bases de datos.

En este punto deberemos acceder a la sección SQL que nos permitirá copiar las sentencias SQL que tenemos en el archivo “localhost” en la raíz de nuestro proyecto. Con ello obtendremos la base de datos y todas las tablas necesarias para la ejecución de nuestra aplicación.

9.2 Configuración

La configuración de la aplicación es muy fácil y cómoda ya que simplemente deberemos cambiar los archivos xml que existen en la carpeta “1htm/xml”.

Existen 2 ficheros que deberemos configurar dependiendo del aula dónde vamos a ejecutar la aplicación y las prácticas que se van a realizar a lo largo del curso.

9.2.1 AULAS

Para la configuración de la distribución de las aulas se deberá editar el fichero "aulas.xml" que se encuentra en la carpeta "xml" dentro de "1htm".

Estructura

La estructura del fichero es la siguiente:

```
<aulas>
  <clase>
    <nombre>B1</nombre>
    <distribucion>MMHMMMRMMHMMMRMMHMMMRMMHMMMRHHHHMMRP</distribucion>
    <puesto>
      <ip>158.42.181.22</ip>
      <maquina>acomp22</maquina>
      <tipo>A</tipo>
    </puesto>
    <puesto>
      <ip>158.42.181.21</ip>
      <maquina>acomp21</maquina>
      <tipo>A</tipo>
    </puesto>
    .
    .
    .
  </clase>
</aulas>
```

Está organizado por etiquetas y simplemente hay que cambiar el contenido de cada etiqueta para configurar el aula.

Significado de las etiquetas

Para cambiar la configuración de las aulas, hay que sustituir el contenido de cada etiqueta por el que queremos mostrar, para ello deberemos conocer que significa cada etiqueta.

<aulas>

Contiene todas las clases disponibles en la aplicación. Actualmente la aplicación sólo está preparada para una clase, pero con unas pequeñas modificaciones del código se podría conseguir adaptar la aplicación para varias clases.

<clase>

Contiene todas las características de las clases.

<nombre>

Aquí podemos cambiar el nombre de la clase, el nombre se visualizará en el listado de clases de la aplicación.

<distribucion>

En esta etiqueta podemos modificar la distribución de la clase. Para ello necesitamos incluiremos la cadena de caracteres correspondiente a la clase. Cada carácter tiene un significado para la aplicación y dependiendo del que introduzcamos se mostrará una distribución u otra.

M -> MESA
H -> HUECO ENTRE MESAS
R -> FIN DE FILA
P -> MESA DEL PROFESOR

<puesto>

Contiene la configuración a cada puesto de trabajo.

<ip>

Cada puesto de trabajo tiene una ip única, por lo que deberemos insertar la ip de cada puesto de trabajo conocer a que ip corresponde cada máquina.

<maquina>

Cada puesto de trabajo también tiene un nombre único, también deberemos escribir el nombre de cada máquina.

<tipo>

Define el tipo de máquina, sólo existen 2 posibilidades:

P -> PROFESOR
A -> ALUMNO

Con unos simples cambios podremos configurar los puestos de trabajo del aula y ver reflejados los cambios en la aplicación.

Cualquier carácter o cambio de etiqueta en el fichero podría dejar sin funcionamiento de la aplicación, por lo que es aconsejable que la edición del fichero sea realizada por el administrador de la aplicación o persona con conocimientos de informática.

9.2.2 PRÁCTICAS

Para la configuración de las prácticas que se realizarán durante todo el curso, se deberá editar el fichero “practicas.xml” que contiene el listado de las prácticas que se quiere evaluar su asistencia y presentación.

Estructura

La estructura del fichero es la siguiente:

```
<temario>
  <practica>
    <codigo>1</codigo>
    <titulo>Practica 1</titulo>
  </practica>
  <practica>
    <codigo>2</codigo>
    <titulo>Practica 2</titulo>
  </practica>
  .
  .
  .
</temario>
```

Significado de las etiquetas

Para cambiar la configuración de las prácticas, hay que sustituir el contenido de cada etiqueta por el que queremos mostrar, para ello deberemos conocer que significa cada etiqueta.

<temario>

Contiene todas prácticas que se van a realizar durante todo el curso y estas se mostrarán en la aplicación.

<practica>

Contiene todas las características de las prácticas.

<codigo>

A cada práctica se le deberá asignar un código numérico, correlativo y único para que se pueda identificar con facilidad.

<titulo>

En esta etiqueta escribiremos el título de la práctica.