UNIVERSIDAD DE MÁLAGA

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA

INGENIERA TÉCNICO/A EN INFORMÁTICA DE GESTIÓN

**GESTIÓN DE UNA ESCUELA INFANTIL VÍA WEB CON CAPACIDAD MULTIPLATAFORMA**

**MANAGING WEB FOR A NURSERY WITH MULTIPLATFORM CAPABILITY**

Realizado por:

Samuel Morales Mangas

Dirigido por:

Antonio J. Nebro Urbaneja

Departamento:

Departamento Lenguajes y Ciencias de la Computación

MÁLAGA, 4 de Diciembre de 2015

UNIVERSIDAD DE MÁLAGA

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA

INGENIERA TÉCNICO/A EN INFORMÁTICA DE GESTIÓN

Reunido el tribunal examinador en el día de la fecha, constituido por:

Presidente/a Dº/Dª. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Secretario/a Dº/Dª. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Vocal Dº/Dª. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

para juzgar el proyecto Fin de Carrera titulado:

**Gestión de una Escuela Infantil vía web con capacidad multiplataforma**

**Managing web for a Nursery with multiplatform capability**

Realizado por D. Samuel Morales Mangas

Tutorizado y dirigido académicamente por D. Antonio J. Nebro Urbaneja,

ACORDÓ POR \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ OTORGAR LA CALIFICACIÓN

DE \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

Y PARA QUE CONSTE, SE EXTIENDE FIRMADA POR LOS COMPARECIENTES

DEL TRIBUNAL, LA PRESENTE DILIGENCIA.

Málaga a \_\_\_\_ de\_\_\_\_\_\_\_\_\_\_\_\_\_\_ del 20\_\_

|  |  |  |
| --- | --- | --- |
| El/La Presidenta | El/La Secretario/a | El/La Vocal |
| Fdo: | Fdo: | Fdo: |

Índice

[**1 Introducción** 5](#_Toc436955842)

[**1.1 Motivación** 5](#_Toc436955843)

[**1.2 Objetivos** 6](#_Toc436955844)

[**1.3 Fases del proyecto** 6](#_Toc436955845)

[**1.4 Contenido de la memoria** 7](#_Toc436955846)

[**2 Tecnologías utilizadas** 8](#_Toc436955847)

[**2.1 Bases de desarrollo** 8](#_Toc436955848)

[**2.2 Desarrollo del Proyecto** 8](#_Toc436955849)

[**2.2.1 Web Service** 9](#_Toc436955850)

[**2.2.2 Sitio Web** 10](#_Toc436955851)

[**2.3 Diseño Web** 15](#_Toc436955852)

[**2.4 Base de datos** 16](#_Toc436955853)

[**2.5 Implementación del BackEnd** 17](#_Toc436955854)

[**2.6 Servidor** 19](#_Toc436955855)

[**2.7 Gestión de BackUp** 20](#_Toc436955856)

[**3 Diseño del proyecto** 22](#_Toc436955857)

[**3.1 Especificacion** 22](#_Toc436955858)

[**3.2 Diseño** 22](#_Toc436955859)

[**4 Implementación del proyecto** 23](#_Toc436955860)

[**4.1 Desarrollo Interfaz Web** 23](#_Toc436955861)

[**4.1.1 Interfaz amigable** 23](#_Toc436955862)

[**4.1.2 Diseño responsive** 23](#_Toc436955863)

[**4.1.3 Conocer la empresa** 23](#_Toc436955864)

[**4.1.4 Oferta de actividades** 23](#_Toc436955865)

[**4.2 Galería fotográfica** 23](#_Toc436955866)

[**4.2.1 Contacto** 23](#_Toc436955867)

[**4.2.2 Redes Sociales** 23](#_Toc436955868)

[**4.3 Evaluación** 23](#_Toc436955869)

[**5 Conclusiones** 25](#_Toc436955870)

[**5.1 Conclusiones** 25](#_Toc436955871)

[**5.2 Propuestas Futuras** 25](#_Toc436955872)

[**6 Apéndices** 27](#_Toc436955873)

[**6.1 Apéndice I: Estructura del CD** 27](#_Toc436955874)

[**6.2 Apéndice II: Índice de ilustraciones** 28](#_Toc436955875)

[**7 Bibliografía** 29](#_Toc436955876)

**1 Introducción**

Actualmente se convive día a día con las tecnologías y las redes de información y poco a poco va surgiendo la necesidad de estar informado de todo en todo momento. A esto no escapa el ámbito familiar y son muchos padres los que piden saber del estado y evolución de sus hijos tanto medicamente como académicamente.

Debido a esta necesidad en auge, son muchas escuelas e instituciones académicas las que aportan o suministran estas herramientas para poder facilitar a los padres y tutores un seguimiento de sus hijos. Y a su vez facilitar a los profesores o educadores la gestión de esta información.

En el rango de edades que comprenden las Escuelas Infantiles es mayor esta necesidad debido a la sensación de indefensión que creen los padres que tienen sus hijos. Y por esos es mayor la necesidad de aportar estas herramientas para poder dar una tranquilidad a los progenitores.

Bajo esta idea, este documento pretende mostrar cómo se genera un sistema de información, para una Escuela Infantil, que pueda cubrir la necesidad de información tanto de los padres como de los profesores y educadores desde un entorno amigable y accesible desde cualquier sitio y momento. Y si es necesario, dando la posibilidad de comunicación a otros sistemas.

## **1.1 Motivación**

El hecho de plantearnos la realización de este proyecto se encuentra en la posibilidad de actualizar y mejorar el sistema de gestión de información que tienen muchas Escuelas Infantiles.

Estos centros, ya sea por dejadez o por no vislumbrar la necesidad de los padres, han ido dejando de lado la gestión de información a tan solo unos determinados anuncios en los tablones de noticias que el centro pone, o por la información que los educadores proporcionan a los padres cuando recogen a sus hijos.

Para mejorar esta situación se plantea un sistema al que puedan acceder los padres y educadores en todo momento y desde cualquier lugar a través de internet. Una página Web que permita a todos los usuarios acceder a la información que necesiten sin importar la plataforma que usen. Y un Web Service que permita el intercambio de información con otros sistemas tanto de instituciones educativas como de otras plataformas de gestión de información.

## **1.2 Objetivos**

El objetivo principal de este proyecto es el diseño e implementación de un Web Service que permita a diferentes sistemas acceder a la información, y un sitio Web que permita a diferentes usuarios visualizar dicha información.

El Web Service debe de:

* Ser rápido en proporcionar la información que se le solicita.
* Ser seguro en el acceso a la información que trata para evitar que ningún sistema o usuario pueda tratar información sensible.
* Tener un protocolo de comunicación que permita a otros sistemas trabajar con él.
* Proporcionar sólo los datos indispensables para evitar fuga de información relevante.

El sitio Web debe de:

* Ser atractiva visualmente para que a los usuarios no les resulte pesada y a los visitantes les interese.
* Adaptar el diseño a los nuevos tiempos para que se posible su correcta visualización en diferentes plataformas.
* Ser intuitiva y fácil de usar para que los usuarios no tengan problemas a la hora de moverse por los diferentes menús.

## **1.3 Fases del proyecto**

El proyecto se ha dividido en las siguientes fases:

* Estudio de los lenguajes a utilizar:
  + HTML5 , PHP y JavaScript para la desarrollo del proyecto.
  + CSS3, Bootstrap y JQueryUI para el diseño de la web.
  + SQL para la implementación de la base de datos.
* Estudio de las tecnologías a utilizar:
  + PhalconPHP para la implementación del BackEnd.
  + Apache como servidor donde se alojará el Web Service y el sitio Web.
  + MySQL como motor de base de datos.
* Diseño e implementación del Web Service y del Sitio Web
  + Especificación de requisitos necesarios para el Web Service y el sitio Web
  + Diseño de la base de datos.
  + Diseño de interfaz de Web Service.
  + Diseño de sitio Web.
  + Desarrollo e implementación del proyecto.

## **1.4 Contenido de la memoria**

Este documento se va a estructurar en varios capítulos, según la cronología de las fases del proyecto:

* El presente capitulo explica las motivaciones y objetivos del proyecto, así como las fases de realización.
* El segundo capítulo explica las distintas tecnologías que se han utilizado para la realización del proyecto, así como los motivos por los que fueran elegidas.
* El tercer capítulo informa del proceso de diseño tanto del Web Service como del sitio Web.
* El cuarto capítulo muestra de toda la implementación del proyecto y una evaluación del mismo.
* En el quinto capítulo se detallan las conclusiones obtenidas con la realización del proyecto y se indican propuestas futuras para mejorar y ampliar el mismo.
* Por último, los capítulos seis y siete contienen los apéndices y bibliografía respectivamente.

**2 Tecnologías utilizadas**

En este capítulo se describen las distintas tecnologías que han sido elegidas para el desarrollo del proyecto, así como la razón por las que se han seleccionado.

## **2.1 Bases de desarrollo**

A la hora de afrontar el desarrollo de este proyecto se han tenido en cuenta diferentes aspectos para poder seleccionar las herramientas más adecuadas:

* **Facilidad**. Se pretende que las herramientas sean lo más fáciles e intuitivas de usar para poder mostrar un desarrollo e implementación claro y ágil.
* **Completo**. A la vista de la gran cantidad de herramientas disponibles, se busca la más completa posible y aquella que permita trabajar tanto en el desarrollo web service como en el sitio web.
* **Rapidez**. Otro factor importante es encontrar las herramientas que puedan automatizar gran parte del desarrollo del proyecto consiguiendo de esta manera más tiempo para pruebas e imprevistos.
* **Actual**. Las herramientas deberán estar actualizadas o tener constancia de que están siendo mantenidas. Así se evitarán problemas de seguridad y aumenta la longevidad del proyecto.

Siguiendo estas bases, las herramientas seleccionadas serán las más adecuadas, completas y actuales que se puedan adaptar a las necesidades del proyecto.

## **2.2 Desarrollo del Proyecto**

El proyecto está dividido en dos secciones importantes y la elección de los lenguajes de desarrollo debe de contemplarse desde dos puntos de vista:

* Por un lado, tenemos el Web Service con el cual se permitirán las comunicaciones entre la base de datos y los usuarios o sistemas que quieran obtener información.
* Por otro lado, el Sitio Web con el cual los usuarios podrán interactuar y disfrutar de los servicios disponibles.

### **2.2.1 Web Service**

A la hora de desarrollar un Web Service hay que comprender que este sistema tiene como objetivo la intención de poder suministrar información a todo aquel que se la solicita mediante una serie de protocolos. Hace la función de intermediario entre la información almacenada en la base de datos y los sistemas o usuarios que solicitan.

Debido a este planteamiento los lenguajes más indicados son aquello que están trabajando en el lado del servidor y no en el del cliente. Por esa razón, y siguiendo las bases, el lenguaje seleccionado es PHP.

**PHP**

Recordando un poco de historia, PHP es el acrónimo de “PHP: Hypertext Preprocessor”. Originalmente fue creado, en 1994, por Rasmus Lerdorf con el nombre de “Personal Home Page Tools” pero actualmente está supervisada por The PHP Group bajo licencia de software libre. Sirve como estándar de facto de PHP.

A lo largo de su historia, este lenguaje ha ido evolucionando y adaptándose a los tiempos para adquirir nuevas capacidades hasta convertirse en un lenguaje de desarrollo muy potente y robusto que está siendo constantemente mantenido por una gran comunidad.



Figura : Logo PHP

Actualmente está en su versión 7, la cual es la que se ha utilizado en este proyecto, y entre las mejoras que trae es un aumento del rendimiento. Pero también tiene otras características que lo hacen atractivo:

* **No tipado**. No es un lenguaje que requiera de la definición de tipos por lo que permite una programación más flexible.
* **Orientado a objetos**. Permite todas las ventajas de la programación con objetos, clases, herencias, etc…
* **Modular**. Permite que se puedan usar el potencial de este lenguaje mediante los módulos necesarios para lo que lo vamos a usar.
* **Retorno de función tipado**. Aunque es un lenguaje flexible a no declarar el tipo de las variables, en esta última versión se permite declarar el tipo del retorno de las funciones para así permitir al desarrollador controlar mejor que va a devolver.

Teniendo en cuenta que este lenguaje es interpretado por el servidor, y ninguna línea de código llega a los navegadores de los usuarios, es muy importante que las cargas de trabajo sean bajas o controladas para que la respuesta del servidor sea lo más rápida posible.

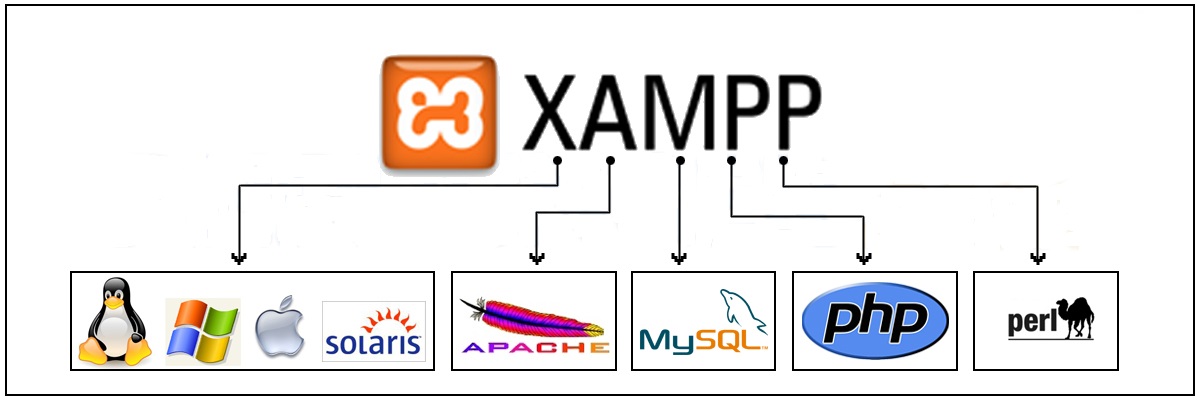


Figura : Herramientas contenidas en XAMPP

Para poder trabajar con este lenguaje se ha decidido utilizar **XAMPP**. Se trata de un entorno de desarrollo multiplataforma (funciona en Windows, Linux y Mac) de software libre con todo lo necesario para trabajar tanto en PHP como en otros lenguajes interpretados. Este entorno incorpora una base de datos MySQL, un servidor Web Apache y los interpretes de lenguaje. Además, siguiendo las bases, está mantenido por una gran comunidad que constantemente está actualizándolo y sacando nuevos plugins para incorporar nuevas herramientas y complementos (como por ejemplo Wordpress, Drupal, Joomla!).

### **2.2.2 Sitio Web**

Para el desarrollo del sitio Web con el que los usuarios podrán visualizar la información que el Web Service les proporcione se utilizarán varios lenguajes diferentes en función de las diferentes partes que se controlarán.

**PHP**

Al estar el sitio Web alojado en un servidor, al igual que el Web Service, es coherente que usemos también este lenguaje para poder implementar las páginas Web que los usuarios van a poder visualizar.

Por otro lado, usaremos este lenguaje para poder interactuar con el Web Service. Será el encargado de:

* Interpretar la información que el usuario está solicitando.
* Realizar la petición al Web Service y recuperar esa información.
* Y finamente interpretarla para poder mostrarla.

Pero además se usarán otros dos lenguajes bastante usuales en desarrollos web: HTML y JavaScript.

**HTML**

Es uno de los lenguajes con más historia que existe y que aún hoy en día sigue siendo primordial en el desarrollo de páginas web y mantenido por una gran comunidad debido a su gran sencillez de manejo y comprensión. Su nombre es el acrónimo en inglés de “Hyper Text Markup Language” y está orientado a un lenguaje básico por etiquetas.



Figura : Logo de HTML

Creado por el físico Tim Berners-Lee en 1990 en el CERN para facilitar a los investigadores la publicación de documentos, ha ido evolucionando y creciendo manteniendo su liderazgo a la hora del desarrollo de la mayoría de las páginas web del planeta hasta el punto de convertirse en un estándar.

Actualmente está supervisado por el organismo W3C (World Wide Web Consortium) encardado de estandarizar casi todas las tecnologías ligadas a la Web.

**HTML5**

Se trata de una evolución del lenguaje HTML, la quinta para aclarar y es el sucesor del estándar HTML4, declarado el lenguaje oficial de la Web en el año 2000. Pero al igual que todos sus antecesores, sigue manteniendo su principal característica, el ser un lenguaje Markup.



Figura : Logo de HTML5.

Esto permite tanto la facilidad de uso para desarrollar como para ser interpretado. De hecho, la mayor parte de los navegadores de internet (Internet Explorer, Firefox, Chrome, Safari…) son compatibles con él y recomiendan tenerlos actualizados a la última versión para disfrutar de todas las posibilidades que ofrece HTML5.

Entre las muchas características que presentan podemos destacar las siguientes:

* **Nuevas etiquetas**. Se han agregado nuevas etiquetas que permiten a los desarrolladores realizar cosas que con versiones anteriores tenían que hacer con otras herramientas.
* **Inserción multimedia**. Se han añadido nuevas etiquetas que permiten incluir y mostrar contenidos multimedia sin apenas complicaciones.
* **Independencia**. Elimina gran cantidad de dependencias que eran necesarias antes mediante los plug-ins para poder conseguir resultados óptimos en las páginas webs.
* **Adaptación**. Ha añadido la capacidad de poder adaptar la web a diferentes plataformas permitiendo, de esta forma, derribar los límites de visualización, que sus antecesores tenía, para que todos los usuarios puedan visualizar páginas web independientemente del sistema que estén usando.



Figura : Características Principales de HTML5

**JavaScript**

Para poder añadir cierta interactividad asíncrona entre el sitio Web y el Web Service se integra y usa este leguaje. Una de las peculiaridades de este lenguaje interpretado es que se usa normalmente en el lado del cliente. Esto significa que la información que obtiene del servidor es tratada en la máquina del cliente, o usuario, permitiendo de esta forma liberar recursos al servidor.

Si unimos esta característica con la tecnología **Ajax** obtenemos una gran particularidad en nuestra página Web. La asincronidad del sitio Web. Por norma general las páginas Webs necesitan enviar una solicitud de información y recargarse con la respuesta del servidor. Esta tecnología Ajax nos permite evitar que la página se recargue al realizar las peticiones asíncronas.

Este lenguaje fue desarrollado originalmente por Brendan Eich que trabajaba en Netscape. En 1995 fue incorporado al navegador Netscape. Pero a su vez empezaron a aparecer diferentes lenguajes similares pero incompatibles como JScript de Microsoft. En 1997 la ECMA (European Computer Manufacturers 'Association) adopto como estándar renombrándolo como ECMAScript.

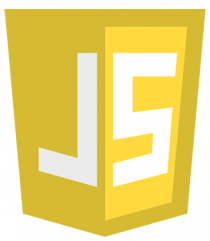


Figura : Logo de JavaScript

Se podría decir que es un dialecto del ECMAScript pero que mantiene muchas de las características de este:

* **Tipado Dinámico**. Es un lenguaje que permite gran flexibilidad al no tener que declarar el tipo de las variables ya que es el contenido de estas la que indica su tipo.
* **Orientado a Objetos**. Está pensado y orientado a trabajar y manipular objetos. Aunque en realidad son arrays asociativos en las que las propiedades de estos “objetos” son clave/valor.
* **Funcional.** Se trabaja principalmente a través de funciones. Estas son las encargadas de tratar la información cada vez que son llamadas por un evento provocado por la página Web.
* **Librerías.** Está mantenido por una gran comunidad de desarrolladores que crean diferentes librerías o paquetes de instrucciones para mejorar aún más el desarrollo de paginas Webs.

**jQuery**

Entre las muchas librerías de JavaScript que podemos encontrar por el mundo de los desarrolladores, una de las más actuales y con más auge es jQuery. Fue presentada en 2006 y rápidamente se popularizo al permitir a los desarrolladores realizar tareas que sin ella necesitarían más tiempo.

Entre muchas de las características que lo hacen tan atractivo están:

* **Elementos DOM.** Permite seleccionar e interactuar con los elementos DOM estandarizados por el W3C (World Wide Web Consortium).
* **Eventos.** Se ejecuta mediante eventos que activan o ejecutan cada una de las instrucciones desarrolladas.
* **Manipulación de CSS.** Permite el cambio de las hojas de estilos CSS y añadir efectos y animaciones.
* **Ajax.** Permite el manejo de instrucciones Ajax y añadir extensiones.

Tras esta breve exposición de los diferentes lenguajes que vamos a usar en el desarrollo del sitio web, se debe informar que estos lenguajes también son soportados por el entorno de trabajo XAMPP expuesto en el apartado anterior.

Además, se usará el editor Sublime Text como herramienta para el desarrollo e implementación de proyectos. Es un editor bastante potente en auge que está siendo mantenido por una comunidad cada vez más en grande de usuarios que van añadiendo poco a poco plug-ins y permitiendo de esta manera unas posibilidades inmensas.

## **2.3 Diseño Web**

Para el sitio Web se eligieron tres lenguajes diferentes para el diseño de tal forma que con ellos pudiéramos cubrir ciertas perspectivas que deben de tener actualmente las páginas web: Estetica amigable, Dinamismo en el contenido y adaptabilidad en las plataformas visuales.

**CSS.**

Son las siglas de “Cascading Style Sheet” y con este lenguaje se consigue definir cómo se va a visualizar o presentar la página web. Se podría decir que con el podemos definir la apariencia estética de cada parte que van a tener las páginas web de nuestro sitio web.

Es un lenguaje que poco a poco ha ido ganando adeptos desde su primera publicación, por el año 1966, hasta llegar a ser uno de los lenguajes más populares y usados en el entorno web. El encargado de establecer sus especificaciones o estandarizar sus “hojas de estilos” es W3C.

Estas “hojas de Estilo”, que nos permiten especificar las reglas por las que se va a regir la presentación de la web, están definidas en un fichero XML aunque no necesariamente se tiene que separar de la estructura de la página web. Es una manera de organizarlo mejor.

**jQueryUI.**

Es un conjunto de librerías para jQuery que añaden a este framework una serie de efectos visuales y plug-ins que permiten a las páginas web que lo usan tener una sensación mayor de dinamismo y frescura.

Esta librería está orientada por módulos de tal forma que se permite aplicar o instalar a nuestro entorno de desarrollo sólo aquellos módulos que nos sean necesarios, consiguiendo de esta forma optimizar los recursos del desarrollo que estemos llevando a cabo.

**Bootstrap.**

Este framework es uno de los más populares actualmente y no hay ninguna pagian web que esté, o se esté en proceso de adaptación, a usar dicha tecnología. En 2011 fue publicado por Twiter como código abierto y desde ese momento se convirtió en uno de los proyectos más populares de GitHub.

Esta herramienta contiene un conjunto de plantillas de diseño con tipografía, formularios, botones, menús de navegación y otros muchos elementos que permiten que nos sea más fácil diseñar y crear una web.

Pero su mayor característica, y donde reside su gran popularidad y potencial, es en la capacidad de adaptabilidad a la interfaz donde se visualizan la web que lo incorporan es su diseño. Esto permite que pueda verse correctamente una página web tanto si se visualiza en un monitor de ordenador, como si se visualiza en un Smartphone.

Esta técnica que incorpora Bootstrap se conoce como “responsive design” y su esencia está en dividir la interfaz en una cuadricula o “Grid” de tal forma que cada elemento que compone la página web que visualizamos tiene un tamaño dentro de esa rejilla.

Este cuadricula está compuesta por 12 columnas que se van adaptando en diferentes tamaños según el interfaz que lo esté visualizando en cada momento. Por tanto, se establecen 4 tipos de tamaño estandarizados: Muy pequeño (xs), Pequeño (sm), Mediano (ms) y Grande (ld).

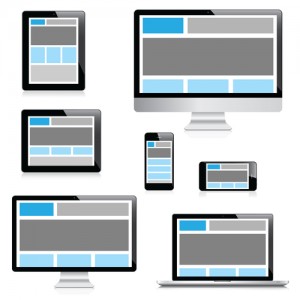


Figura : Adaptación con Bootstrap

## **2.4 Base de datos**

**SQL**

Al igual que en cualquier proyecto, se necesita almacenar la infamación que los usuarios quieren visualizar o tratar desde sus sistemas clientes. Esta información es necesario almacenarla en un sistema que sea capaz de catalogarla y tratar según las necesidades del proyecto.

Para eso es necesario un sistema gestor de base de datos relacional que permita gestionar dicha información con un lenguaje fácil y rápido. Este lenguaje es SQL (Structured Query Language) creado con el propósito de especificar diversos tipos de operaciones en bases de datos relacionales, fue estandarizado en 1986.

Es un lenguaje declarativo que permite realizar cálculos algebraicos y relacionales con la información que se almacena en las bases de datos además de recuperarla o alterarla.

**MySQL**

Para este proyecto se ha seleccionado una de las bases de datos más populares y usadas que existen en la actualizad. MySQL se creó en 1995 por Michael Widenius y un grupo de colaboradores.



Figura : Logo de MySQL

A lo largo de su historia ha ido evolucionando para adaptarse a los tiempos y para aportar a los usuarios y desarrolladores múltiples características que iban demandando.

Algunas de las características que tienen encontramos:

* Multiplataforma. Se han ido creando diversas versiones apra que puedan adaptarse a los diferentes sistemas que existen.
* Seguridad. Ofrece conectividad segura para trabajar con la información.
* Replicación. Soporta la capacidad de realizar una copia exacta del sistema, o parte del el, para salvaguardar las estructuras de base de datos y su contenido.

## **2.5 Implementación del BackEnd**

Para el tratamiento de la información por detrás de la parte visual es necesario que se utilicen herramientas o Frameworks que estén ligados al lenguaje de programación principal que se usará. En este caso PHP.

Actualmente existen muchos Framework diferentes con sus puntos fuertes y débiles, pero una de las características que debe de cumplir dicha herramienta es la velocidad y optimización de sus procesos.

Se busca esa característica ya que se usará, para este proyecto, en dos entornos diferentes pero comunes a la hora de tratar la información, y se podría llegar a ralentizar bastante los recursos. Por esa razón se pretende que el Framework este optimizado al máximo sin perder la liberta de la que tanto goza PHP.

Para el lenguaje PHP existen múltiples opciones de herramientas, pero es PhalconPHP la que se ha destacado por su optimización a bajo nivel de sus estructuras y funciones para aplicaciones MVC (Modelo-Vista-Controlador).



Figura : Logo de Phalcon

Publicado en 2012 por Andrés Gutiérrez, es relativamente moderno, pero gracias a su gran atractivo ha ido creciendo poco a poco hasta tener una gran comunidad de usuarios que lo mantienen y lo engrandece, consiguiendo que en 2015 saliera la versión 2.0.

Es contradictorio que una herramienta para PHP se haya desarrollado en C pero esto le ha permitido bajar el nivel de desarrollo de sus funciones principales y conseguir optimizar el rendimiento y evitar el tiempo que se tarda si fuera un lenguaje interpretado.

Entre las muchas características que podemos encontrar en Phalcon están:

* Contiene una amplia librería de Request que permiten un acceso a datos muy optima gracias a un uso propio de ORM permitiendo, de esta forma, migrar desde PostgreSQL a MySQL o entre otros motores de base de datos.
* Motor de plantillas que permite a los desarrolladores generar fácil y rápidamente formularios y páginas webs. Dicho motor de plantillas es Volt.
* Almacenamiento en Cache para un acceso más rápido a los datos que se van a tratar.
* Acceso directo a las estructuras internas de PHP para optimizar cada ejecución.

Se puede entender Phalcon, a la hora de trabajar con él, como una extensión de PHP que al estar hecho en C, como ya hemos dicho, permite que su instalación y manejo sean rápidos y modulares, además de fáciles de extrapolar. Esto lo podemos entender con la figura 10 en la que se puede ver con claridad cómo está conectado Phalcon con un proyecto cualquiera.

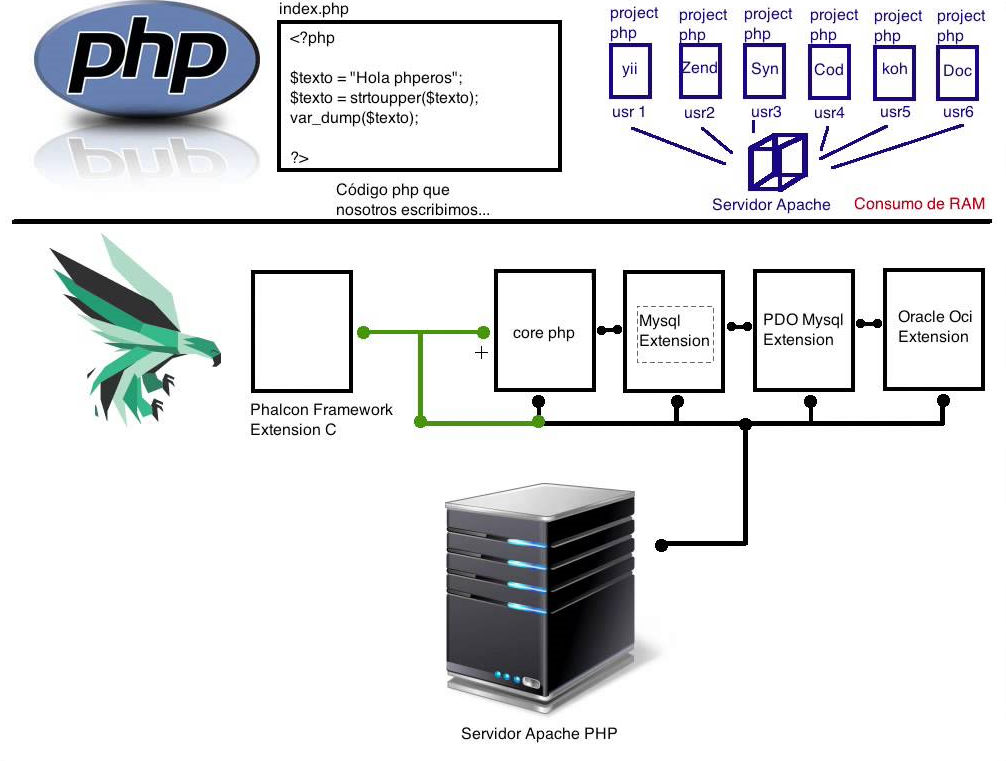


Figura : Esquema de un proyectocon Phalcon.

## **2.6 Servidor**

Tanto el Web Service como el sitio Web que se desarrollaran necesitan de un servidor donde se puedan alojar los archivos de ambos proyectos. Aunque no necesariamente deben de estar en un mismo servidor, para el desarrollo del mismo sí se realizará así para la comodidad del alumno.

De entre todos los posibles servidores que existen actualmente se ha seleccionado Apache porque es un servidor de código abierto que tiene compatibilidad con múltiples plataformas (Unix, Microsoft, Macintosh…) y permite sitios virtuales.

Se empezó a desarrollar en 1995 y a lo largo de todo este tiempo ha sido mantenido y evolucionado por una gran comunidad de usuarios supervisados por Apache Software Foundation. Esto le ha permitido ser uno de los servidores más utilizados tanto en el ámbito profesional como en el académico.

Entre sus muchas características (por las cuales también se ha seleccionado) podemos encontrar:

* **Altamente configurable.** Tiene una gran colección de atributos que le permiten al usuario adaptar y casi personalizar el servidor a las necesidades del proyecto.
* **Seguridad.** Al tener una gran comunidad de usuarios que lo van manteniendo, se ha estado resolviendo rápidamente todas las vulnerabilidades que has estado apareciendo.
* **Modular.** Consta de un núcleo y de múltiples módulos que añaden funcionalidad.

## **2.7 Gestión de BackUp**

Llegados a este punto necesitamos una herramienta que nos permita salvaguardar toda la información del proyecto mientras se realiza el desarrollo y la implementación del mismo. Es por eso que se optó por una herramienta fácil de usar como GitHub.

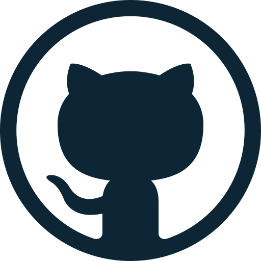


Figura : Logo GitHub

Realmente GitHub es una plataforma de desarrollo colaborativo muy popular en la comunidad de desarrolladores actualmente y está en funcionamiento desde 2010. Esta plataforma trabaja usando el sistema de control de versiones Git. De esta manera conseguimos el sistma de backup que necesitamos y una comunidad de usuarios capaz de mantener y aportar ideas a nuestros proyectos.

El sistema Git fue desarrollado por Linus Torvalds queriendo establecer un sistema de control de versiones que mantuviera la eficiencia y confiabilidad de dichos backup cuanto los desarrollos tuvieran un gran número de archivos.



Figura : GitHub y Git

Algunas de sus muchas características son:

* **Desarrollo no lineal.** Permite realizar el desarrollo de un proyecto usando la técnica del “colash” en la que se avanza por etapas que no necesariamente tienen que esperar a terminar una para empezar otra.
* **Gestión distribuida.** Permite que cada uno de los desarrolladores o colaboradores tengas una copia del código para poder avanzar.
* **Rapidez de gestión.** Permite y facilita la gestión de proyectos grandes al poder manipular rápidamente las diferencias entre ficheros de los diferentes desarrolladores.

**3 Diseño del proyecto**

En este capítulo se mostrarán los diferentes procesos que se llevaron a cabo para diseñar el proyecto que pudiera cumplir con las expectativas de los padres y empleados de una Escuela Infantil.

Teniendo en cuenta que la Escuela Infantil que se elige como modelo no tiene informatizado ningún aspecto de su gestión, es necesario empezar desde cero en el tratamiento tanto de los requisitos que debe de cumplir el sistema como en diseño del sitio web.

Y hay que indicar que el proyecto está compuesto por dos sistemas diferentes pero conectado y, en cierto sentido, similares. Uno es el Web Service, que establecerá la comunicación entre la base de datos y los sistemas externos. Y el otro es el sitio web que hará de interfaz con los usuarios.

Estas dos partes en las que se compone el proyecto permite la modularización del sistema y por tanto la facilidad para que una empresa que tenga varios centros educativos pueda escalar y agrandar su negocio.

## **3.1 Especificación**

Para poder realizar un diseño correcto del proyecto, antes es necesario recopilar y analizar las necesidades y requisitos que debe de cumplir dicho sistema final.

* **Modularidad**

Para eso hay que entender que uno de los puntos esenciales, dicho antes, es dar la posibilidad de crecimiento de la empresa sin que ello repercuta en el tratamiento de la gestión de la misma.

Con esto se pretende que la empresa no tenga problemas si quiere expandirse, pero sin perder el control de la información sensible de la que dispone.

* **Centralización de datos**

El sistema debe de estar controlado y supervisado para que ningún dato sensible pueda salir o ser usado por usuarios ajenos o fraudulentos.

* **Seguridad y confidencialidad**

El sistema debe de tener unos sistemas de seguridad para que los accesos a la información no se realicen por usuarios que no deben visualizarla.

* **Comunicación**

El sistema debe de tener la posibilidad de proporcionar información a cualquier usuario o sistema externo que lo solicite.

* **Fácil**

El sistema debe de ser lo más intuitivo posible para los usuarios a la hora de su manejo.

Siguiendo estas necesidades básicas se plantea un sistema de Web Service que permita la comunicación tanto de sistemas externos, que quieran asociarse con la empresa, como otros sistemas externos que permitan a los usuarios obtener la información que necesitan.

Además, se plantea un sitio Web que permita a los usuarios interactuar con dicho Web Service de forma fácil, cumpliendo con la posibilidad de que, a su vez, dicho sitio web pueda cumplir con la labor de ser la web oficial de la empresa.

Planteando este proyecto en dos sistemas hay que indicar las especificaciones que debe de cumplir cada uno de ellos.

### **3.1.1 Web Service**

aquí hablamos de los requisitos del web service

### **3.1.2 Sitio Web**

aquí hablamos de los requisitos de la web

## **3.2 Diseño**

En esta sección se podrá mostrar el diseño planteado de cada uno de los dos sistemas que componen el proyecto, así como las diferentes interrelaciones que existen entre ambos.

### **3.2.1 Web Service**

aquí hablamos del diseño y diagramas del web service

### **3.2.2 Sitio Web**

aquí hablamos del diseño y diagramas de la web

**4 Implementación del proyecto**

**4.1 Desarrollo Interfaz Web**

**4.1.1 Interfaz amigable**

**4.1.2 Diseño responsive**

**4.1.3 Conocer la empresa**

**4.1.4 Oferta de actividades**

**4.2 Galería fotográfica**

**4.2.1 Contacto**

**4.2.2 Redes Sociales**

**4.3 Evaluación**

Tras toda la implementación del proyecto, se planteó realizar una serie de pruebas que permitieran evaluar y depurar los diferentes aspectos que conforman el sistema. Aunque ya se había estado probando en cada una de las fases por las que pasaba el proyecto y cada uno de los sistemas desarrollados, se veía necesario realizar pruebas generales.

Para eso se comprobaron:

* El correcto funcionamiento de cada una de las partes desarrolladas del Web Service tanto con las peticiones realizadas desde el sitio Web, como desde una interfaz independiente (previamente adaptada al protocolo de comunicación)
* El correcto funcionamiento de cada una de las partes desarrolladas en el sitio Web usando diferentes navegadores como Chrome, Firefox, Internet Explorer o Edge..

**5 Conclusiones**

En este capítulo se mostrarán las conclusiones a las que se ha llegado tras la finalización de dicho proyecto.

## **5.1 Conclusiones**

Con la realización de este proyecto se ha podido generar un sistema compuesto de dos partes entrelazadas y con capacidad de extrapolarse fácilmente.

Gracias a las tecnologías y herramientas seleccionadas se ha conseguido un desarrollo limpio y sencillo. La gran trayectoria que tienen todas estas herramientas permite que sean robustas y estén bien cohesionadas entre ellas de tal forma que a los usuarios nos resulte fácil su utilización.

La cantidad de documentación que existe de los lenguajes seleccionados y de la activa comunidad que participa en los foros de los frameworks elegidos, ha permitido que este desarrollo e implementación del proyecto haya sido sencillo y muy instructivo.

La realización de este proyecto ha permitido aumentar y adquirir nuevos conocimientos sobre las distintas tecnologías que actualmente se extienden por el mundo tecnológico. Y ha inspirado al estudiante a seguir mantenido la misma senda de aprendizaje para mejorar sus conocimientos y afrontar nuevos proyectos con los mismos o retos diferentes.

## **5.2 Propuestas Futuras**

Tras terminar el proyecto siempre va apareciendo nuevas funcionalidades o mejoras para añadir que aumentan y engrandecen el proyecto poco a poco. Alguna de estas ideas que se plantean son:

* **Multi-idioma.** Dar la posibilidad al sitio Web la capacidad de visualizarse en otros idiomas.
* **Mejorar API.** Mejorar la API del Web Service para aumentar el número de instrucciones con las que puede interactuar con otros sistemas.
* **Script de Web.** Mejorar el rendimiento del sitio Web y los scripts que en ellos se encuentran.
* **Apps para Moviles.** Realizar una aplicación móvil que permita interactuar con el Web Service.

Estas propuestas se plantean para futuras actualizaciones del proyecto que unos mismo y otros desarrolladores puedan diseñar e implementar.

**6 Apéndices**

## **6.1 Apéndice I: Estructura del CD**

En el CD que se adjunta a este documento se puede encontrar la siguiente estructura:

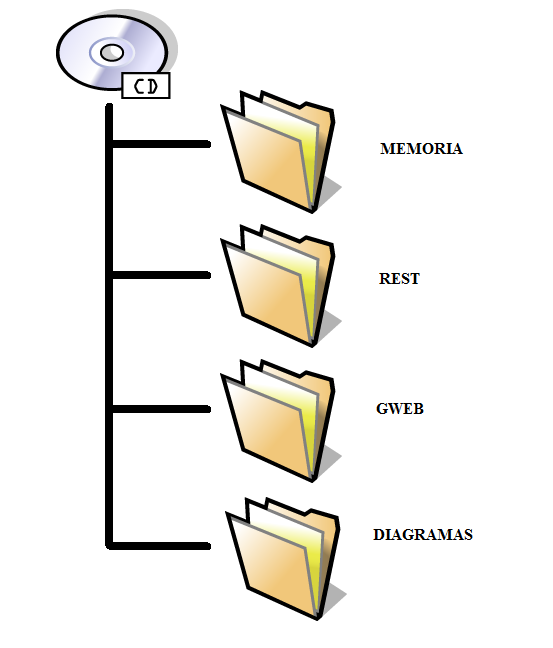


Figura : Estructura de CD

MEMORIA: Memoria del proyecto en formato pdf y docx.

REST: Código fuente del sitio REST.

GWEB: Código fuente del sitio Web.

DIAGRAMAS: Archivos de todos los diagramas que se han realizado en …….

## **6.2 Apéndice II: Índice de ilustraciones**

[Figura 1: Logo PHP 9](#_Toc436955889)

[Figura 2: Herramientas contenidas en XAMPP 10](#_Toc436955890)

[Figura 3: Logo de HTML 11](#_Toc436955891)

[Figura 4: Logo de HTML5. 12](#_Toc436955892)

[Figura 5: Características Principales de HTML5 12](#_Toc436955893)

[Figura 6: Logo de JavaScript 13](#_Toc436955894)

[Figura 7: Adaptación con Bootstrap 16](#_Toc436955895)

[Figura 8: Logo de MySQL 17](#_Toc436955896)

[Figura 9: Logo de Phalcon 18](#_Toc436955897)

[Figura 10: Esquema de un proyectocon Phalcon. 19](#_Toc436955898)

[Figura 11: Logo GitHub 20](#_Toc436955899)

[Figura 12: GitHub y Git 21](#_Toc436955900)

[Figura 13: Estructura de CD 27](#_Toc436955901)

**7 Bibliografía**

[1] Página Web con tutoriales de HTML5, CSS, Bootstrap. <http://www.w3schools.com>

[2] Página Web del W3C Consortium. [www.w3.org/html](http://www.w3.org/html)

[3] Web oficial de PhalconPHP. <https://www.phalconphp.com/es/>

[4] Página web oficial de PHP. [www.php.net](http://www.php.net)

[5] Web oficial de MySql. <https://www.mysql.com>

[6] Web oficial de jQuery. <https://jquery.com>

[7] Página Web oficial de jQueryUI. <https://jqueryui.com>

[8] Web oficial de Bootstrap. <http://getbootstrap.com>

[9] Web con tutoriales de PhalconPHP, PHP, jQuery y HTML5. <http://uno-de-piera.com/>

[10] Página Web oficial de GitHub. <https://github.com/>

[11] Web oficial de XAMPP. <https://www.apachefriends.org/es/index.html>

[12] Página Web de Wikipedia. <https://es.wikipedia.org>