

UNIVERSIDAD DE MÁLAGA
ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA

INGENIERO/A 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, Diciembre de 2015

UNIVERSIDAD DE MÁLAGA

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA

INGENIERO/A 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.....	1
1.1 Motivación	1
1.2 Objetivos	2
1.3 Fases del proyecto.....	2
1.4 Contenido de la memoria.....	3
2 Tecnologías utilizadas	5
2.1 Bases de desarrollo	5
2.2 Desarrollo del Proyecto.....	5
2.2.1 Web Service	6
2.2.2 Sitio Web	7
2.3 Diseño Web	13
2.4 Base de datos.....	15
2.5 Implementación del BackEnd	17
2.6 Servidor	19
2.7 Gestión de BackUp.....	19
3 Diseño del proyecto	21
3.1 Especificación	21
3.1.1 Web Service	23
3.1.2 Sitio Web	23
3.2 Diseño	25
4 Implementación del proyecto	29
4.1 Desarrollo de Web Service.	29
4.2 Desarrollo Interfaz Web	39
4.3 Evaluación.....	45
5 Conclusiones	47
5.1 Conclusiones	47
5.2 Propuestas Futuras	47
6 Apéndices	49
6.1 Apéndice I: Estructura del CD	49
6.2 Apéndice II: Índice de ilustraciones	51

6.3 Apéndice III: Índice de Tablas.....	53
6.4 Apéndice IV: Matriz de requisitos.....	55
6.5 Apéndice V: Diagramas de casos de Uso.....	65
6.6 Apéndice VI: Escenarios de Casos de Uso.....	71
6.7 Apéndice VII: Sitio Web	87
7 Bibliografía.....	93

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 y 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 ello también 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 fuera necesario, creando la opción 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 sea 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 el 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 capítulo 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 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 la soliciten.

Debido a este planteamiento los lenguajes más indicados son aquellos 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 1: Logo PHP

Actualmente está en su versión 7, que es la que se ha utilizado en este proyecto, y entre las mejoras que aporta figura un aumento de su rendimiento. Aunque 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.

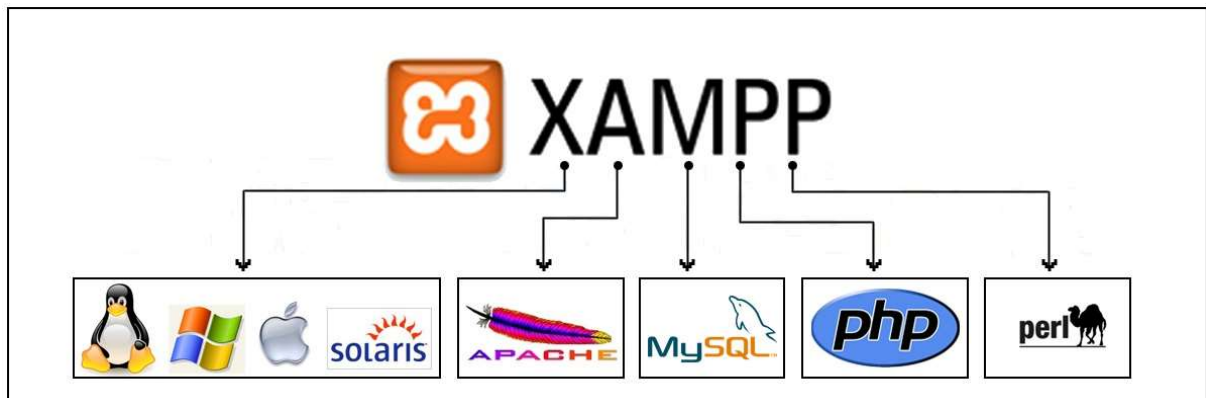


Figura 2: Herramientas contenidas en XAMPP

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 plug-ins 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 finalmente 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 3: 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) encargado 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 4: 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 5: Características Principales de HTML5

JavaScript

Para poder añadir cierta interactividad asíncrona entre el sitio Web y el Web Service se integra y usa este lenguaje. 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 asincronicidad 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 6: 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 el 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 páginas Webs.

jQuery

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

Entre las muchas 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 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: Estética 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, allá 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 la gran mayoría de las páginas web usan o están en proceso de adaptación a dicha tecnología. En 2011 fue publicado por Twitter 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 en 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 cuadrícula o “Grid” de tal forma que cada elemento que compone la página web que visualizamos tiene un tamaño dentro de esa rejilla.

Este cuadrícula 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 7: Adaptación con Bootstrap

2.4 Base de datos

SQL

Al igual que en cualquier proyecto, se necesita almacenar la información 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 tratarla 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 actualidad. MySQL fue creado en 1995 por Michael Widenius y un grupo de colaboradores.



Figura 8: 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 se iban demandando.

Entre algunas de las características que tienen, encontramos:

- **Multiplataforma.** Se han ido creando diversas versiones para 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 libertad 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 9: 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.

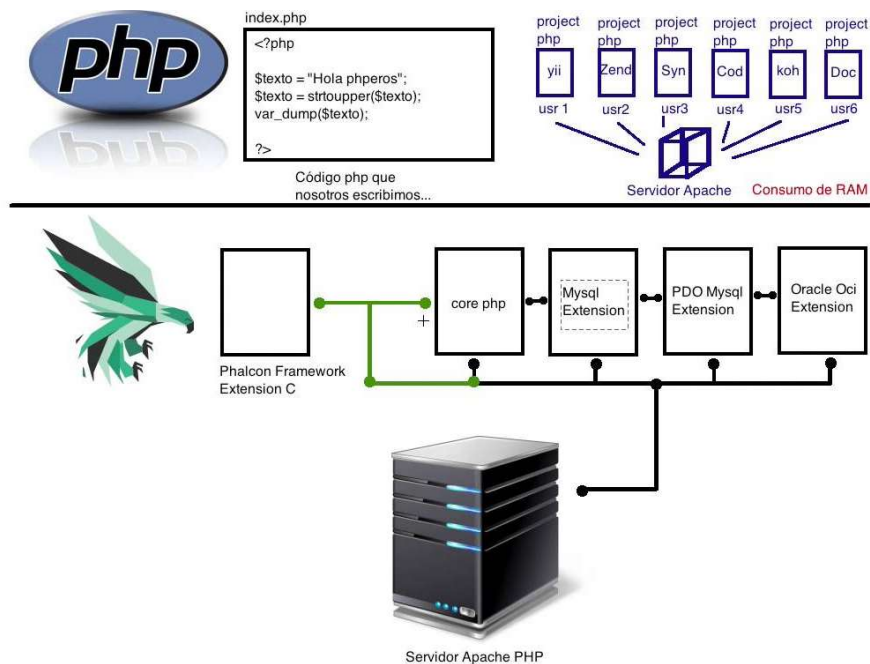


Figura 10: Esquema de un proyecto con Phalcon.

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 han estado resolviendo rápidamente todas las vulnerabilidades que han ido 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 11: 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 sistema 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 12: 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 con objeto de 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, ya mencionado, 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

Con vistas a un proyecto creado para una empresa real se plantean una serie de requisitos que condicionarán el diseño de esta parte del proyecto. A continuación, se puede ver algunos de los requisitos, aunque el resto de ellos los podremos encontrar en el apéndice III.

Tipo	Descripción
No Funcional	Obtener la mayor productividad del negocio
No Funcional	Minimizar costes de los recursos
No Funcional	Minimizar el impacto en la transición a la nueva era
No Funcional	La mayor parte de las transmisiones de datos sean telemáticas
No Funcional	Facilidad de uso por parte de los empleados y los padres
No Funcional	Maximizar la automatización de los procesos de la empresa
No Funcional	Minimizar el tráfico de información no automatizada
No Funcional	Minimizar la dependencia de terceras empresas
No Funcional	Facilitar el trabajo de los empleados con las tecnologías instaladas
No Funcional	Mantener la legislación vigente
No Funcional	Agilizar el trabajo diario a los empleados

Tabla 1: Tabla de Requisitos para Web Service

3.1.2 Sitio Web

Por otra parte, es importante saber cuáles son las especificaciones principales que debe cumplir la página web para satisfacer las necesidades de los futuros usuarios. A continuación, se muestra una lista de algunos requisitos, aunque la totalidad de ellos los podemos ver en el Apéndice III.

Tipo	Descripción
Funcional	Hacer una página web de la escuela infantil.
Funcional	Los padres podrán acceder a la web.
Funcional	En la web se podrá ver la información de los avances de los alumnos, sólo por sus padres y sus profesores.
Funcional	En la web se podrá ver el proyecto educativo de la empresas.
Funcional	En la web se podrá ver el funcionamiento de los centros y las características de cada uno de ellos.
Funcional	En la web debe aparecer la información del personal del centro y sus fotos para poder identificarlas.
Funcional	En la web debe aparecer información de actividades realizadas o previstas.
Funcional	También debe de aparecer una sección de contactos que figuren la dirección de los centros y las formas de contactar con ellos.

Tabla 2: Tabla de requisitos para Web

3.2 Diseño

En esta sección se muestra el diseño planteado del proyecto, mediante un diagrama de casos de uso, para cubrir gran parte de las necesidades y requisitos que se vieron en el apartado anterior.

El diagrama que se muestra a continuación pretende hacer ver la sección esencial que todo proyecto de gestión MVC (Modelo-Vista-Controlador) debe de tener y que se solicita entre sus requisitos. Esta sección es el mantenimiento de cada uno de los usuarios o elementos que se tratan en la empresa, además de la gestión de usuarios.

Pero hay que tener en cuenta que cada una de estas partes que componen esta sección tiene una implementación específica en cada uno de los sistemas que compone este proyecto. De esta manera tanto el Web Service como el sitio Web tendrán unas funciones definidas que aplicarán según los eventos que se produzcan o las acciones que los usuarios ejecuten.

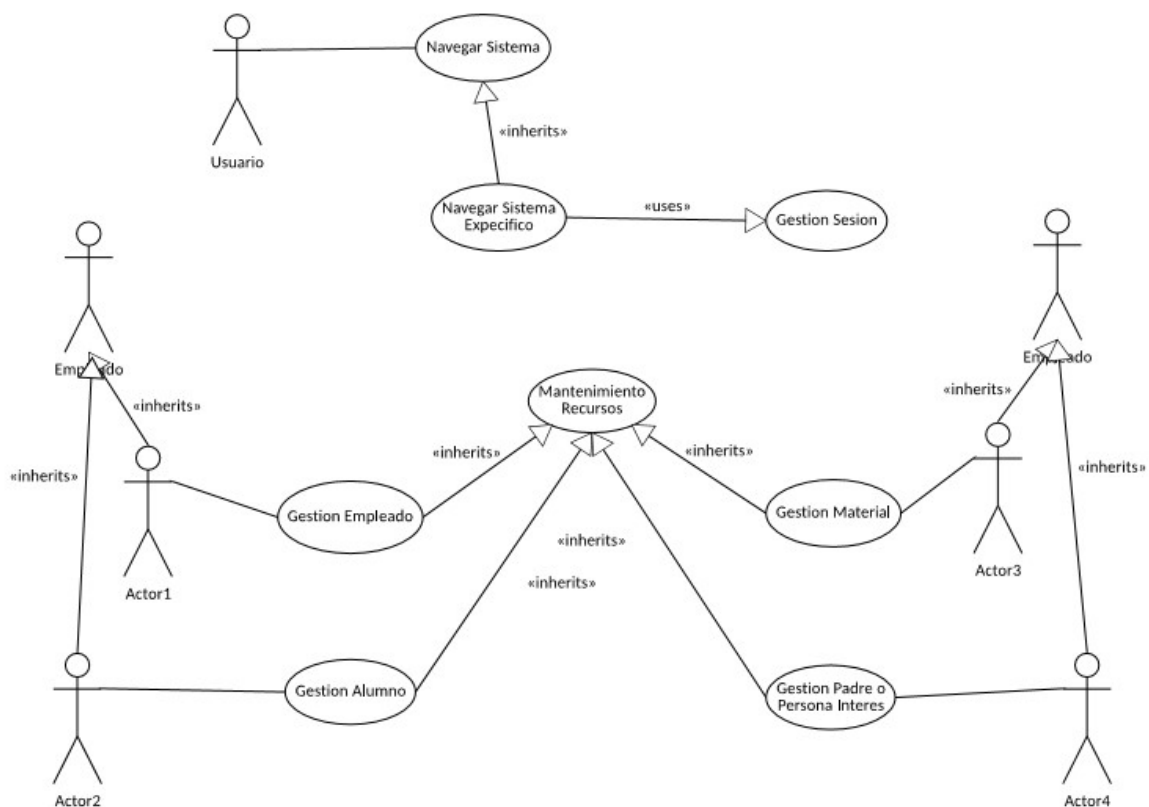


Figura 13: Diagrama de Casos de Uso Mantenimiento.

Se podrán ver el resto de diagramas de casos de uso con mayor detalle en el Apéndice IV donde se podrá visualizar el resto de las secciones que son necesarias para la gestión completa de una Escuela Infantil.

Teniendo ya el diagrama de caso de uso, debemos de entender cuáles son los casos de uso que afectan a cada parte del sistema que se está desarrollando y cuál es su función en dicho proyecto.

Nombre	Navegar sistema
Casos de uso incluidos	Ninguno
Actores	Usuario
Objetivo	La navegación de cualquier usuario no logueado o invitado
Escenarios Básico	<ul style="list-style-type: none"> • el usuario selecciona las opciones del menú • el sistema gestiona sesión • el sistema recupera la información necesaria • el sistema muestra la información solicitada al usuario
Sistemas afectados	Sitio Web

Tabla 3: Navegar sistema

Nombre	Navegar sistema específico
Casos de uso incluidos	Ninguno
Actores	Usuario
Objetivo	Navegar por el sitio web una vez que el usuario se ha logueado
Escenarios Básico	<ul style="list-style-type: none"> • el usuario selecciona las opciones específicas del menú • el sistema gestiona sesión • el sistema recupera la información específica • el sistema muestra la información solicitada
Sistemas afectados	Sitio Web

Tabla 4: Navegar sistema específico

Nombre	Gestión de sesión
Casos de uso incluidos	Ninguno
Actores	Usuario
Objetivo	Realizar los procesos en el sistema para controlar cuales son los usuarios que acceden y el tiempo
Escenarios Básico	<ul style="list-style-type: none"> • el sistema comprueba identidad de usuario • comprueba permisos de usuario
Sistemas afectados	Web Service

Tabla 5: Gestión de sesión

Nombre	Mantenimiento Recursos
Casos de uso incluidos	Ninguno
Actores	Empleado
Objetivo	Realizar los procesos con la base de datos para gestionar la información que se le indica por la interfaz
Escenarios Básico	<ul style="list-style-type: none"> • El empleado accede con su usuario • Gestión de sesión • selecciona el recurso que quiere administrar • Da de alta, actualiza o elimina el recurso
Sistemas afectados	Web Service

Tabla 6: Mantenimiento Recursos

Nombre	Gestión Empleados
Casos de uso incluidos	Ninguno
Actores	Empleado
Objetivo	Gestionar el mantenimiento de los empleados
Escenarios Básico	<ul style="list-style-type: none"> • El empleado accede con su usuario • Gestión de sesión • Da de alta, actualiza o elimina a empleados
Sistemas afectados	Web Service y Sitio Web

Tabla 7: Gestión Empleados

Nombre	Gestión Alumno
Casos de uso incluidos	Ninguno
Actores	Empleado
Objetivo	Gestionar el mantenimiento de los alumnos.
Escenarios Básico	<ul style="list-style-type: none"> • El empleado accede con su usuario • Gestión de sesión • Da de alta, actualiza o elimina a alumnos
Sistemas afectados	Web Service y Sitio Web

Tabla 8: Gestión Alumno

Nombre	Gestión Material
Casos de uso incluidos	Ninguno
Actores	Empleado
Objetivo	Gestionar el mantenimiento de los materiales usados en la empresa
Escenarios Básico	<ul style="list-style-type: none"> • El empleado accede con su usuario • Gestión de sesión • Da de alta, actualiza o elimina a materiales
Sistemas afectados	Web Service y Sitio Web

Tabla 9: Gestión Material

Nombre	Gestión Padres o Responsables
Casos de uso incluidos	Ninguno
Actores	Empleado
Objetivo	Gestionar el mantenimiento de los padres y responsables de alumnos.
Escenarios Básico	<ul style="list-style-type: none"> • El empleado accede con su usuario • Gestión de sesión • Da de alta, actualiza o elimina a padres o personas de interés para los alumnos
Sistemas afectados	Web Service y Sitio Web

Tabla 10: Gestión Padres o Responsables

4 Implementación del proyecto

Ya llegados a este capítulo, vamos a describir cual ha sido el proceso de implementación del mismo para poder cumplir las necesidades y requisitos planteados en capítulos anteriores.

Como se ha indicado, el proyecto se plantea como un conjunto de dos sistemas que interaccionan entre ellos para poder cumplir con las funciones que se le solicita. Por esa razón, este capítulo se va a desarrollar en dos partes. Una que visualice el proceso del Web Service, y el otro que mostrará el proceso para el sitio Web.

El hecho de explicarlo de esta manera nos hará comprender mejor cómo se ha desarrollado el proyecto y, además, podremos entender que la decisión de dividirlo de esta manera hace que se cumplan varios requisitos (entre ellos la centralización de información, la escalabilidad y la interacción con sistemas externos).

4.1 Desarrollo de Web Service.

A la hora de afrontar el desarrollo del Web Service hay que entender que la función principal del mismo es poder dar la información que oferta a sus solicitantes en el menor tiempo y lo más clara posible.

Se podría interpretar que estamos hablando de una página web o de una aplicación de escritorio, pero no es eso ya que este tipo de programas está pensado para interactuar con usuarios o personas física a la que hay que mostrarle la información. Y sin embargo un Web Service está pensado para dar la información que otro sistema le solicita. De tal manera que un Web Service solamente se comunicará con otras máquinas.

Para que esa comunicación sea correcta hay que establecer un estándar o protocolo que tanto el sistema externo como el Web Service utilicen para poder hablar. Una especie de idioma. A esto se le denomina API (*Application Programming Interface*).

API

Para establecer esta API se establece primero como se iba a realizar las comunicaciones. En este caso se iban a realizar a través de internet, pero la información se mandaría por POST. Un método de enviar la información que queda oculto a la vista de los usuarios que navegan por internet.

La información que se enviaría por este método se tendría que añadir en una variable de tal forma que tanto el emisor como el receptor sepan dónde encontrar la información que necesitan.

```
$data = array('dato' => $dato //“Información que se va a transmitir”  
            , 'status' => $status  
            , 'message' => $message);
```

Tabla 11: Estructura de Datos

Pero hay que entender que no toda la información se puede enviar por internet tal cual. Por red solo se pueden enviar un conjunto de caracteres. Por esa razón, si trabajamos con orientación a objetos, hay que establecer un protocolo de transformación que permita convertir cualquier cosa de nuestro programa en una cadena de caracteres.

Para poder hacer esto PHP nos proporciona una serie de herramientas que permiten tanto transformar un objeto en cadena de caracteres, como codificar la información para enviarla por internet. Usaríamos las funciones “Serialize/Unserialize” para transformar un dato en cadena de caracteres y la función “Json_Encode” para codificar la información en un tipo Json para enviar por internet.

```
$dato = serialize($dato);  
$data = array('dato' => $dato  
            , 'status' => $status  
            , 'message' => $message);  
$json = json_encode($data);
```

Tabla 12: Codificación de Datos

Seguridad

Por otro lado, hay que tener en cuenta que la información que se va a transmitir contiene datos sensibles o confidenciales. Así que hay que establecer, también un protocolo de seguridad en la transmisión de esta información.

Para ello lo que haremos será codificar el dato que se transmite, pero no el “status” ni el mensaje ya que, por ejemplo, si por alguna razón se recibe un paquete de error, porque no tiene privilegios y otra razón, no hace falta decodificar la información ya que sólo tendremos que observar el campo “status”.

Para codificar la información existen muchos algoritmos, pero Phalcon nos proporciona uno bastante potente que lo trae integrado en la clase “Crypt”. El algoritmo es “blowfish” y es un

codificador de bloques simétrico que usa bloques de 64 bits con claves de este 32 a 448 bits. Es un algoritmo bastante efectivo creado por Bruce Schneier en 1993 y desde entonces se ha estado utilizando en muchísimos conjuntos de codificadores y productos de cifrado.

En nuestro caso, para estandarizarlo hemos creado la clase “Encriptador” que realiza la encriptación y des-encriptación de la información de forma automática. Y como clave hemos establecido la fecha en la que se trasmite la información. Esto nos permite tener una clave actualizada y cambiante en cada momento.

```
class Encriptador extends ClaseAbstracta
{
.....
    $cripto = new Crypt();
    $cripto->setCipher('blowfish');
    $this->putCripto($cripto);
    $key = date('d/m/y');
    $this->putKey($key);
.....
}
```

Tabla 13: Clase Encriptación

RESTFul

Las características que debe de cumplir nuestro Web Service son las de una aplicación RESTFul. Este tipo de aplicaciones son sistemas hipermedia distribuidos que proporcionan recursos que pueden ser accedidos desde la red utilizando un identificador Uniforme de recursos.

El término de este tipo de aplicaciones se originó por Roy Fielding en el 2000 en su tesis doctoral. Actualmente se utiliza este término para describir cualquier interfaz entre sistemas que utilice directamente HTTP para obtener datos o indicar la ejecución de operaciones sobre los datos.

Las características que debe de cumplir son las siguientes:

- **Protocolo cliente/servidor sin estado**

Cada mensaje HTTP contiene toda la información necesaria para comprender la petición. Como resultado, ni el cliente ni el servidor necesitan recordar ningún estado de las comunicaciones entre mensajes.

- **Operaciones bien definidas**

Se suelen utilizar acciones claras y concisas para indicar que debe de hacer la aplicación. Las más importantes suelen ser POST, GET, PUT y DELETE.

- **Uso de hipermedios.**

Tanto para la información de la aplicación como para las transiciones de estado de la aplicación: la representación de este estado en un sistema REST son típicamente HTML o XML.

Para conseguir que nuestra aplicación Web Service tenga las características de las aplicaciones RESTful hemos creado tantos ficheros como nos sean necesarias en el mantenimiento de la información o clases y las hemos metido en una carpeta (la carpeta “rest”) para tenerlas controladas.

La estructura de estos ficheros debe de tener tantas funciones como instrucciones podamos realizar para el tratamiento de esa información. En la siguiente tabla se puede ver un ejemplo de las instrucciones para tratar los Roles.

```

<?php

* Obtener listado de roles
*/
$app->post('/api/roles/getlist', function() use ($app)
{
    //Obtenemos el JSON que se ha enviado
    $data = $app->request->getJsonRawBody();

    //Desmontamos el JSON
    $dato = $data->dato;

    //Desmontamos los datos de envio
    $datoenvio = new Datoenvio();
    $datoenvio->obtenerDatos($dato);

    //Obtenemos la Sesion y la informacion
    $sesion = $datoenvio->getSesion();
    $usuario = $datoenvio->getDato();

    //Tratamos la Sesion y la informacion
    //Primero comprobamos la sesion
    if($sesion->checkearEstado() == true)
    {
        //Si es correcta tratamos la informacion
        //Obtenemos listado de usuarios
        $rol = new Rol();
        $arrayRoles = $rol->getListado();

        //Devolvemos mensaje correcto
        $status = 'OK';
        $message = 'Listado de rol.';

        //Montamos los datos de envio
        $dato = $datoenvio->enviarDatos($sesion, $arrayRoles);
    }
    else
    {

```

```

        //Cerramos sesion
        $sesion->cerrarSesion();

        $status = 'ERROR-1';
        $message = 'Sesion caducada.';

        //Montamos los datos de envio
        $dato = $datoenvio->enviarDatos($sesion, $usuario);
    }
    //Montamos el JSON
    $data = array('dato' => $dato
        , 'status' => $status
        , 'message' => $message);

    //Codificamos el JSON
    $json = json_encode($data);

    //Enviamos el JSON
    return $json;
});

```

Tabla 14: Ejemplo de fichero RESTful

Sesión

Para terminar, tenemos que hacer que el Web Service tenga un control de las sesiones para que los diferentes usuarios no interfieran en las acciones o información que está tratando otro usuario o sistema.

Para ello estableceremos un protocolo de tratamiento de sesiones con el cual cada vez de un sistema o usuario establezca conexión quede registrado en el sistema. Así mantendremos un histórico de los accesos al Web Service. Esto nos ayudara también a tener una seguridad del sistema.

Para establecer este protocolo crearemos una clase “Sesión” que controlara tanto la creación de un identificador, por cada usuario que se logué o usuario invitado, como el mantenimiento de estos identificadores que deberán de caducar a los 10 minutos de inactividad. Una vez pasado este tiempo se deberá loguear de nuevo.

```
class Sesion extends ClaseAbstracta
{
.....
$session = new Sesiones();
$session->numerosesion = $numerosesion;
$session->usuario = $usuario->id;
$session->fechaactual = date('Y/m/d_H:i:s');
$session->fechacaducidad = date('Y/m/d_H:i:s',mktime(date('H'), date('i') + 10, 0,
date('m'), date('d'), date('Y')));
.....
}
```

Tabla 15: Clase Sesión

Teniendo en cuenta que este identificador de sesión está asignado a cada usuario cuando hay comunicación, se establece que es información sensible y por tanto ira codificada. De tal forma que tendremos que cambiar el protocolo de transmisión de datos ya que habrá que codificar tanto la información como la sesión. Para eso creamos una clase llamada “Datoenvio”. Con esta clase automatizamos tanto la seguridad como la codificación de los datos.

```

class Datoenvio extends ClaseAbstracta{
.....

    public function enviarDatos($sesion, $dato)
    {
        $this->putSesion($sesion);
        $this->putDato($dato);

        //Seralizamos el objeto
        $serial = new Serializador();
        $info = $serial->serializar($this);

        //Encriptamos la informacion
        $encrip = new Encriptador();
        $envio = $encrip->encriptar($info);

        return $envio;
    }
.....
}

```

Tabla 16: Clase Datoenvio

Servicios

Para finalizar, tenemos que hacer que nuestro sistema esté capacitado para responder a las peticiones que se le solicitan en todo momento. Para ello tenemos que hacer que el Web Services este escuchando en todo momento y eso lo conseguimos con la configuración del servidor Apache que trae por defecto.

Y, por otro lado, una vez que llegan las llamadas desde internet al servidor, y este las redirige a la carpeta donde esta nuestro proyecto, es aquí donde tenemos que configurar las carpetas a las que se tiene acceso. El código necesario lo podemos ver en la tabla 17.

```
$di = new FactoryDefault();

$di->set('url', function () use ($config) {
    $url = new UriResolver();
    $url->setBaseUri($config->application->baseUri);

    return $url;
}, true);
```

Tabla 17: Configuración de URL

Con estas bases podemos desarrollar ya el resto del Web Service creando todas las clases que nos permitan interactuar con la base de datos y todas las demás clases que nos permitan tanto procesar la información como almacenarla para enviársela al usuario que la espera.

4.2 Desarrollo Interfaz Web

Para realizar la página Web, al igual que cualquier otra, hay que desarrollar dos partes. Una tratará la información (BackEnd), y la otra, que será la parte visual, hará de interfaz con los usuarios (FrontEnd).

Para cada una de las partes hay que enfocarlo de forma distinta ya que el BackEnd es la encargada de tratar la información que se mostrará, además de ser la responsable de realizar las peticiones al Web Service, y de recibir la respuesta de este. Sin embargo, el FrontEnd simplemente se encarga de visualizar y mostrar correctamente la información intentando que la estética sea atractiva para los usuarios y la funcionalidad sea lo más intuitiva posible.

BackEnd

En las páginas Web que usualmente se suelen hacer, esta parte se encarga de realizar la conexión con la base de datos y buscar la información que necesita para mostrársela a los usuarios mediante el FrontEnd. Pero en esta situación el servidor Web no tiene una base de datos a la que pueda acceder. Por el contrario, tiene que hacer peticiones al Web Service esperando la respuesta. Este tipo de desarrollo nos evita tener que configurar la conexión de base de datos y el generar los ficheros de “modelos” de datos necesarios. Aunque la mecánica es parecida ya que seguimos teniendo que realizar una petición y también tener las clases de los datos que vamos a recibir.

Es por ello que en esta parte podremos copiar tanto las clases de los tipos de datos que se van a enviar desde el Web Service, como las clases necesarios para la transmisión de la información. Es decir, todo lo necesario para mantener el protocolo de comunicación que habíamos establecido en el Web Service.

Para establecer la conexión con el Web Service se usarán los controladores que normalmente se usan si estuviéramos accediendo a una base de datos. En el modelo MVC, son los controladores los que establecen la logia de acceso a la base de datos para indicar que información necesitan. En este caso establecerá la petición al Web Service. En la tabla 18 se puede ver el código de ejemplo que tiene el controlador “Usuario” para recuperar la información.

```

class UsuarioController extends ControllerBase
{
    .....
    public function registerAction()
    {
        if($this->request->isPost())
        {
            $email = $this->request->getPost('nombre');
            $password = $this->security->hash($this->request->getPost('password'));

            $usuario = new Usuario();
            $usuario->putUsuario($email, $password, 1, 1);

            $sesion = new Sesion();
            $sesion->putSesion(0, $email, 1, date('Y/m/d_H:i:s'),
            date('Y/m/d_H:i:s',mktime(date('H'), date('i') + 10, 0, date('m'), date('d'), date('Y'))));

            $datoenvio = new Datoenvio();
            $dato = $datoenvio->enviarDatos($sesion, $usuario);

            $data = array('dato' => $dato
                        , 'status' => 'TO_REGISTER'
                        , 'message' => 'Realizar registro.');
```

```

            $json = json_encode($data);

            $url = 'http://localhost/rest/api/usuarios/registrar/';

            $opciones = array('http' => array('method' => "POST"
                        , 'header' => 'Content-type: application/json'
                        , 'content' => $json
                        , 'timeout' => 60)
                );

            $contexto = stream_context_create($opciones);

            $json = file_get_contents($url, false, $contexto);

```

```

        $data = json_decode($json);

        $dato = $data->dato;

        $datoenvio->obtenerDatos($dato);

        $sesion = $datoenvio->getSesion();
        $usuario = $datoenvio->getDato();

        if($data->status == 'OK')
        {
            $this->_registerUsuario($usuario, $sesion);
            $this->flash->success('Bienvenido ' . $usuario->getEmail());
            return $this->forward('index/index');
        }

        $this->session->remove('auth');
        $this->flash->error('Error en nombre/password');
    }

    return $this->forward('index/index');
}
.....
}

```

Tabla 18: Controlador Usuario

FrontEnd

Para el desarrollo de la parte visual de la página Web tenemos que afrontarla pensando en los requisitos que se indicaron para el desarrollo del proyecto. Entre ellos se pide que sea ágil, intuitiva y estar adaptada a los nuevos tiempos. Con estas premisas se usan las herramientas indicadas en el capítulo 2.2.2.

Para empezar hay que idear un boceto del aspecto que se pretende que tenga la página Web. En este sentido se pretende que dicha Web sea minimalista para que los usuarios no se vean sobrecargados por la cantidad de elementos en pantalla.

Así que se ha ideado que en la página Web haya cuatro secciones bien definidas que se puede ver en la figura 14:

- **Cabecera.**

En esta sección se mostrará el nombre de la escuela infantil y datos de la dirección donde se puede encontrar.

- **Pie**

Esta sección está dedicada para informar de los copyright y otros derechos de autor.

- **Menú**

Esta sección está preparada para que los usuarios puedan navegar por los diferentes contenidos que pueden mostrarse en el cuerpo.

- **Cuerpo**

En esta sección se visualizarán los diferentes contenidos que se pueden mostrar en la página web sobre la escuela infantil.

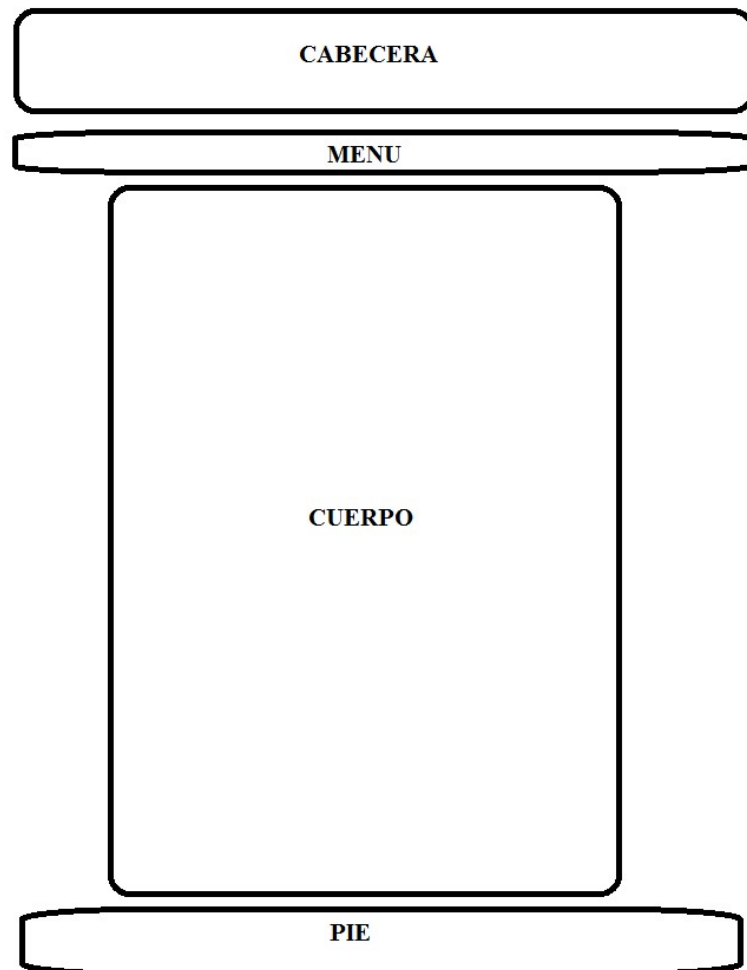


Figura 14: Esquema de Sitio Web

Una vez que ya hemos definido el esqueleto que debe de tener la página Web, tenemos que adaptarla para que tenga la capacidad “responsive”. De esta manera permitiremos que la página Web pueda verse sin problemas en cualquier plataforma que visualice Internet. Para ello tenemos que utilizar componentes de Bootstrap en el código de nuestra paginas como por ejemplo la que se muestran en la tabla 19.

```
<div class="btn-group">
  <button type="button" class="btn btn-default">Left</button>
  <button type="button" class="btn btn-default">Middle</button>
  <button type="button" class="btn btn-default">Right</button>
</div>
```

Tabla 19: Ejemplo de grupo de botones.

También se utilizarán elementos dinámicos para mostrar la información que el usuario quiere ver y que resulten más fácil de manejar y amigables a la vista. Para ello utilizaremos las librerías jQueryUI como se ha indicado en el capítulo 2.2.2.

Uno de los elementos primordiales, y que ayudan mucho a diferenciar las secciones, es el Vertical Tabs Functionality que se pueden ver en la imagen 15.

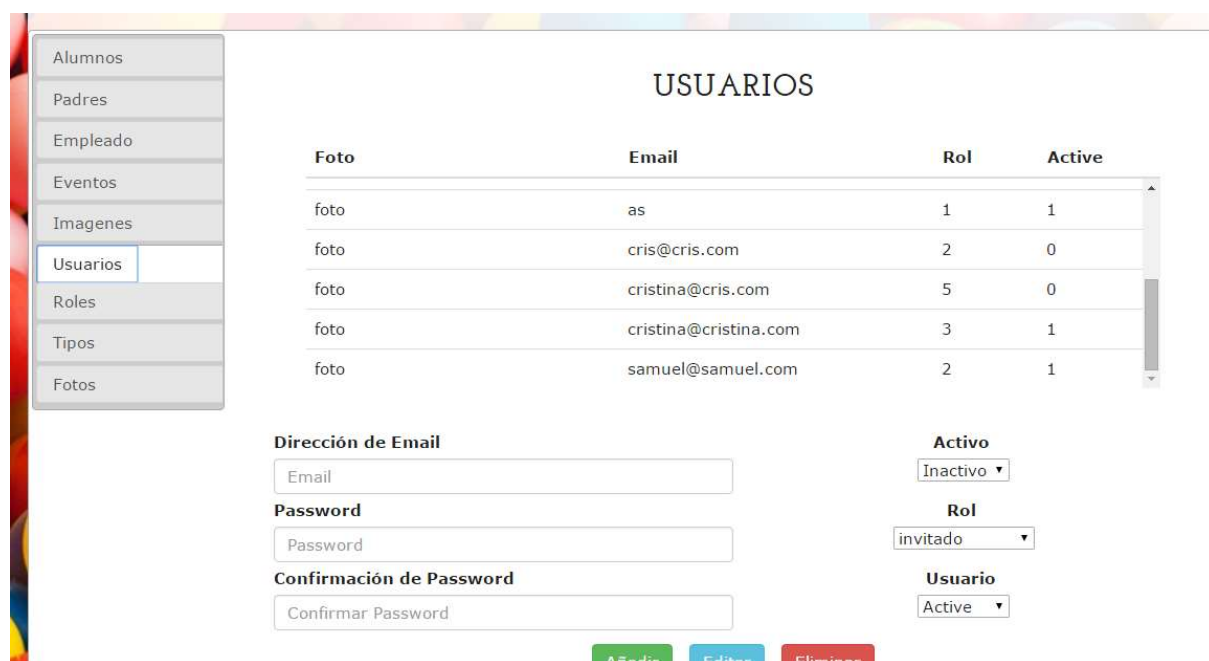


Figura 15: Visualización del Vertical Tabs

Ya para terminar hay que definir las diferentes secciones que se mostraran en el cuerpo de la página web y que ayudaran a los usuarios a poder ver la información que estén buscando. Estas secciones se mostrarán en la sección “Menú” del boceto de la web mediante una clausula “nav” para que así pueda ser “responsive”.

- **Home**

En ella se mostrará una pequeña presentación de la escuela infantil dando información de los ideales que tiene la empresa.

- **Nosotros**

En esta sección se explican con mayor claridad cuáles son los ideales de la empresa, como surgió y cuáles son los intereses que la mantienen con la actitud tan positiva.

- **Blog**

Esta sección está pensada para que los usuarios puedan dejar sus comentarios y experiencias acerca de la escuela infantil.

- **Contacto**

Aquí se muestra toda la información necesaria para poder ponerse en contacto con la empresa o para poder localizarla.

- **Trabajo**

Aunque esta sección tiene el nombre de “Trabajo”, realmente es una sección polivalente que está adaptada al perfil del usuario que entre. Dependiendo del tipo de usuario tendrá más opciones o menos. Por eso en esta sección es donde deben de estar la mayor parte de los elementos que agilizan el trabajo o la navegación de los usuarios.



Figura 16: Barra de menú

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 manteniendo 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 van apareciendo nuevas funcionalidades o mejoras para añadir, y que aumentan y engrandecen el proyecto poco a poco. Algunas 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.
- **Secciones.** Terminar el resto de las secciones necesarias para que todos los perfiles disponibles en una escuela infantil puedan trabajar en el sitio Web.

- **Script de Web.** Mejorar el rendimiento del sitio Web y los scripts que en ellos se encuentran.
- **Apps para Móviles.** 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:

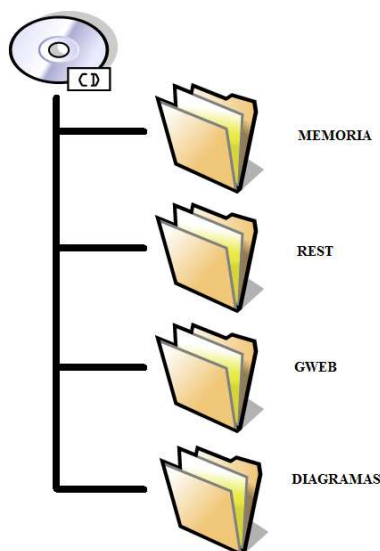


Figura 17: Estructura de CD

MEMORIA: Memoria del proyecto en formato pdf y docx. También podemos encontrar el documento del Anteproyecto.

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.....	6
Figura 2: Herramientas contenidas en XAMPP	7
Figura 3: Logo de HTML	8
Figura 4: Logo de HTML5.	9
Figura 5: Características Principales de HTML5	9
Figura 6: Logo de JavaScript.....	10
Figura 7: Adaptación con Bootstrap.....	14
Figura 8: Logo de MySQL	15
Figura 9: Logo de Phalcon	17
Figura 10: Esquema de un proyecto con Phalcon.....	18
Figura 11: Logo GitHub	20
Figura 12: GitHub y Git	20
Figura 13: Diagrama de Casos de Uso Mantenimiento.....	25
Figura 14: Esquema de Sitio Web	43
Figura 15: Visualización del Vertical Tabs	44
Figura 16: Barra de menú	45
Figura 17: Estructura de CD	49
Figura 18: Caso de Uso de almacén 1	65
Figura 19: Caso de Uso de almacén 2	66
Figura 20: Caso de Uso de cocina	67

Figura 21: Caso de Uso de Profes	68
Figura 22: Caso de Uso de Padres	69
Figura 24: Pagina Home	87
Figura 25: Pagina Nosotros.....	88
Figura 26: Pagina Blog	89
Figura 27: Pagina Contacto.....	90
Figura 28: Pagina usuario logueado y en su menu.....	91
Figura 29: Pagina Home Responsive	92

6.3 Apéndice III: Índice de Tablas.

Tabla 1: Tabla de Requisitos para Web Service.....	23
Tabla 2: Tabla de requisitos para Web	23
Tabla 3: Navegar sistema	26
Tabla 4: Navegar sistema específico	26
Tabla 5: Gestión de sesión.....	27
Tabla 6: Mantenimiento Recursos	27
Tabla 7: Gestión Empleados.....	27
Tabla 8: Gestión Alumno	28
Tabla 9: Gestión Material	28
Tabla 10: Gestión Padres o Responsables	28
Tabla 11: Estructura de Datos	30
Tabla 12: Codificación de Datos	30
Tabla 13: Clase Encriptación.....	31
Tabla 14: Ejemplo de fichero RESTful	34
Tabla 15: Clase Sesión	35
Tabla 16: Clase Datoenvio	36
Tabla 17: Configuración de URL	37
Tabla 18: Controlador Usuario	41
Tabla 19: Ejemplo de grupo de botones.	43
Tabla 20: Tabla de Requisitos	64

Tabla 21: Tabla de Escenarios de Casos de Uso..... 86

6.4 Apéndice IV: Matriz de requisitos.

ID	Autor	Tipo	Descripción	Prioridad	Estado
EVS 1.1			<i>Estudio de la Solicitud</i>		
1	Director de empresa	No Funcional	Informatizar la empresa completamente	Alta	Propuesto
2	Director de empresa	No Funcional	Cumplir LOPD	Alta	Propuesto
3	Director de empresa	No Funcional	Cumplir leyes de educación	Alta	Propuesto
4	Director de empresa	No Funcional	Obtener la mayor productividad del negocio	Media	Propuesto
5	Director de empresa	No Funcional	Minimizar costes de los recursos	Media	Propuesto
6	Director de empresa	No Funcional	Minimizar el impacto en la transición a la nueva era	Baja	Propuesto
7	Director de empresa	No Funcional	La comunicación entre las secciones de la empresa sea rápida y clara	Media	Propuesto
8	Director de empresa	No Funcional	La comunicación entre los empleados y la empresa sea rápida y clara	Media	Propuesto
9	Director de empresa	No Funcional	La comunicación entre los empleados y los padres sea rápida y clara	Alta	Propuesto
10	Director de empresa	No Funcional	La mayor parte de las transmisiones de datos sean telemáticas	Media	Propuesto
11	Director de empresa	No Funcional	Facilidad de uso por parte de los empleados y los padres	Alta	Propuesto
12	Director de empresa	No Funcional	Automatización de los procesos logísticos y financieros	Alta	Propuesto

EVS 2.1			<i>Valoración del Estudio de la Situación Actual</i>		
13	Subdirector de centro	No Funcional	Tener un sistema de información para poder interactuar con la plataforma Séneca de la Consejería de Educación de la Junta de Andalucía.	Media	Propuesto
EVS 3.1			<i>Identificación de las Directrices Técnicas y de Gestión</i>		
14	Director de empresa	No Funcional	Maximizar la automatización de los procesos de la empresa	Media	Propuesto
15	Director de empresa	No Funcional	Minimizar el tráfico de información no automatizada	Media	Propuesto
16	Director de empresa	No Funcional	Minimizar la dependencia de terceras empresas	Alta	Propuesto
17	Director de empresa	No Funcional	Facilitar el trabajo de los empleados con las tecnologías instaladas	Media	Propuesto
18	Director de empresa	No Funcional	Mantener la legislación vigente	Alta	Propuesto
19	Director de empresa	No Funcional	Agilizar el trabajo diario a los empleados	Media	Propuesto
EVS 3.2			<i>Identificación de Requisitos</i>		
20	Director de empresa	Funcional	Hacer una página web de la escuela infantil.	Alta	Propuesto
21	Director de empresa	Funcional	Los padres podrán acceder a la web.	Alta	Propuesto
22	Director de empresa	Funcional	En la web se podrá ver la información de los avances de los alumnos, sólo por sus padres y sus profesores.	Alta	Propuesto
23	Director de empresa	Funcional	En la web se podrá ver el proyecto educativo de la empresas.	Alta	Propuesto
24	Director de empresa	Funcional	En la web se podrá ver el funcionamiento de los centros y las características de cada uno de ellos.	Alta	Propuesto
25	Subdirector de centro	Funcional	Los padres podrán tener acceso a las fotos que se realicen en las fiestas, eventos puntuales y rutina diaria.	Alta	Propuesto
26	Subdirector de centro	Funcional	En la web debe aparecer la información del personal del centro y sus fotos para poder identificarlas.	Alta	Propuesto

27	Director de empresa	Funcional	En la web deben aparecer los precios y tarifas.	Media	Propuesto
28	Subdirector de centro	Funcional	En la web debe aparecer información de actividades realizadas o previstas.	Alta	Propuesto
29	Director de empresa	Funcional	También debe de aparecer una sección de contactos que figuren la dirección de los centros y las formas de contactar con ellos.	Alta	Propuesto
30	Subdirector de centro	Funcional	Y también debe de aparecer en la web una sección de ultimas noticias.	Alta	Propuesto
31	Subdirector de centro	Funcional	Se debe de informar a los padres de cualquier actividad prevista o de cualquier noticia mediante una circular o aviso. (se realizara por email, SMS o wasap)	Alta	Propuesto
32	Subdirector de centro	Funcional	Tablón de anunciar informatizado, que estará en el centro, donde los padre puedan consultar las notificaciones que ahí se cuelguen.	Media	Propuesto
33	Director de empresa	Funcional	Proporcionar, mediante la web o la app (aplicación de móvil), un servicio de ludoteca, canguro o peluquería.	Media	Propuesto
34	Subdirector de centro	Funcional	Realizar una aplicación móvil donde se permita a los padres tener acceso a la información de sus hijos.	Alta	Propuesto
35	Educadora	Funcional	Notificar a los padres, mediante la app, de noticias, actividades, alertas o cualquier otra cosa.	Alta	Propuesto
36	Director de empresa	Funcional	Permitir a los padres, mediante la app, realizar cambios en la recogida de sus hijos indicando quien va a recogerlos y usando una clave de confirmación para esa persona nueva.	Media	Propuesto
37	Cocinera	Funcional	El sistema informático debe avisar cuando se necesite reponer cualquier producto de cocina, limpieza o material de la empresa.	Alta	Propuesto
38	Educadora	Funcional	El sistema informático debe avisar (por la web, email o app) a los padres de la reposición de material de sus hijos (pañales, muda, toallitas, agua, leche, material, etc.).	Media	Propuesto
39	Subdirector de centro	No Funcional	El sistema de información debe de tener un sistema de reconocimiento de personas para poder permitir la recogida de uno de los alumnos.	Media	Propuesto

40	Educadora	Funcional	El sistema informático avisara a la clase donde se encuentre el alumno de que debe prepararse porque vienen a recogerlo. Y también de la persona que viene a recogerlo.	Media	Propuesto
41	Educadora	Funcional	El sistema informático debe de avisar a la educadora, o responsable del alumno, de los cambios que soliciten los padres, mediante email o app, tanto en la recogida como en la alimentación u otros temas.	Alta	Propuesto
42	Director de empresa	No Funcional	El sistema informático debe de llevar la gestión financiera de la empresa y la gestión con la Consejería de Educación de la Junta de Andalucía.	Media	Propuesto
43	Subdirector de centro	No Funcional	El sistema de información debe permitir a los padres implicarse en la educación de sus hijos aportando recursos, o formación, a la empresa. Para ello debe de permitir que los padres avisen mediante la web, o la app, a la dirección.	Baja	Propuesto
44	Cocinera	No Funcional	Se debe llevar un control del almacén de material y de los alimentos para poder hacer previsiones de reposición.	Media	Propuesto
45	Cocinera	No Funcional	Las dietas de alimentación deben estar certificadas por un médico.	Alta	Propuesto
46	Cocinera	Funcional	El sistema debe tener un control de la dieta de cada niño y aconsejar al cocinero por si el alumno tiene alguna alergia.	Media	Propuesto
47	Cocinera	No Funcional	El sistema debe de avisar y aconsejar a la cocinera si los padres han decidido cambiar la dieta del alumno.	Media	Propuesto
48	Cocinera	Funcional	El sistema debe de tener un control de los alimentos para poder certificar la calidad de los mismos.	Alta	Propuesto
49	Cocinera	Funcional	El sistema debe de tener controlado la cantidad de alimentos que hay de cada tipo, el estado de conservación y aconsejar al cocinero su utilización en la dieta o retirada del almacén para su destrucción.	Media	Propuesto
50	Subdirector de centro	No Funcional	El sistema debe controlar los pedidos, tanto de comidas como de materia, mediante los albaranes correspondientes para registrarlos y notificarlos a diferentes secciones de la empresa. (almacén, cocina o contabilidad).	Alta	Propuesto

51	Subdirector de centro	Funcional	El albarán de cada pedido se pasara o a almacén o a cocina, y la factura a contabilidad.	Alta	Propuesto
52	Cocinera	No Funcional	Con los pedidos de alimentos, será la cocinera quien dé el visto bueno a los productos y después será ella quien registre el albarán en cocina y pasara la factura a subdirección para registrarlo en el sistema y pasarlo a contabilidad.	Media	Propuesto
53	Cocinera	No Funcional	Sera la cocinera la que registre los alimentos en el sistema cuando los almacene en la cocina.	Media	Propuesto
54	Subdirector de centro	No Funcional	Con los pedidos de material, será la subdirectora o cualquier educadora la que dará el visto bueno del material recibido y de su confirmación con el albarán. Le pasará tanto el albarán como la factura a subdirección y será esta la encargada de registrarlos en el sistema. Para pasarlos después a contabilidad.	Media	Propuesto
55	Subdirector de centro	No Funcional	Sera la educadora o la subdirección la encargada de registrar los materiales en el sistema cuando los almacene.	Media	Propuesto
56	Educadora	Funcional	Debe de haber una megafonía en cada aula para avisar de cualquier incidencia a las educadoras.	Media	Propuesto
57	Director de empresa	Funcional	Cada centro debe de tener un sistema de conexión inalámbrico.	Media	Propuesto
58	Director de empresa	Funcional	Cada personal de la empresa debe de disponer de una Tablet sincronizada con la red para poder acceder a la app de la empresa y así notificar, o consultar, cualquier incidencia de los alumnos o la empresa.	Media	Propuesto
59	Director de empresa	Funcional	El acceso a las sesiones sensibles de la web será mediante usuario y clave.	Alta	Propuesto
60	Director de empresa	Funcional	El acceso a la app será mediante usuario y clave.	Alta	Propuesto
61	Director de empresa	Funcional	Los padres y empleados podrán acceder a los mismos contenidos tanto en la web como en al app.	Media	Propuesto
62	Subdirector de centro	Funcional	Los padres solo podrán acceder a contenido de sus hijos y a los anuncios de la empresa y el centro.	Alta	Propuesto
63	Subdirector de centro	Funcional	Los educadores solo podrán acceder a contenido de la empresa, sus datos y los datos de los alumnos de los que son responsables.	Alta	Propuesto

64	Subdirector de centro	No Funcional	Las notificaciones del cambio de la persona encargada de recoger al alumno serán escarizadas y solo podrá hacerlas los padres.	Alta	Propuesto
65	Subdirector de centro	No Funcional	Al inicio del curso, los padres deben de informar de las posibles personas que recogerán a los alumnos. Y deben de traer la información necesaria para que luego puedan ser identificados.	Alta	Propuesto
66	Subdirector de centro	Funcional	Cualquier otra persona que no esté identificada entre los posibles recogedores (y que tenga que recoger al alumno de forma extraordinaria), solo los padres podrán informar de dicha persona y deberán aportar o el DNI, o foto o el número de móvil. El sistema lo registrara a la espera de una futura confirmación presencial de los padres. Tras al recogida se le notificara a los padres.	Alta	Propuesto
67	Subdirector de centro	No Funcional	La subdirección tendrá acceso o privilegios a todas las áreas de su centro.	Media	Propuesto
68	Director de empresa	No Funcional	La dirección tendrá acceso o privilegios a todas las áreas de todos los centros y resto de la empresa.	Media	Propuesto
69	Director de empresa	No Funcional	Cada responsable de área tendrá acceso exclusivo a su sección.	Media	Propuesto
70	Director de empresa	Funcional	Todos los sistemas de información estarán centralizados en un servidor.	Alta	Propuesto
71	Director de empresa	Funcional	Todos los centros deben tener conexión a internet.	Alta	Propuesto
72	Director de empresa	Funcional	Cada centro tendrá un sistema de apoyo que permita hacer de enlace entre las tabletas de los empleados y el sistema centrar.	Alta	Propuesto
73	Director de empresa	Funcional	Los sistemas de apoyo también harán labores de servidor por si la conexión con el sistema centrar se pierde.	Alta	Propuesto
74	Director de empresa	No Funcional	Los tabloneros de anuncios digitales solo informarán de los anuncios de sus respectivos centros o, en su caso, de alguna notificación general.	Media	Propuesto
ASI 2.1			<i>Obtención de Requisitos</i>		
75	Director de empresa	No Funcional	Los usuarios que estén trabajando deben de ver el nombre de su usuario para tener constancia de estar dentro del sistema.	Media	Propuesto

76	Director de empresa	No Funcional	Los usuarios sólo podrán tener acceso a los sectores o web a los que tengan permiso según su categoría.	Media	Propuesto
77	Director de empresa	No Funcional	Habrà un menú en el que figurarán los diferentes sectores a los que tiene acceso un usuario.	Media	Propuesto
78	Director de empresa	No Funcional	El sistema debe de avisar a los responsables, mediante notificación en terminal, de cualquier percance.	Media	Propuesto
79	Director de empresa	No Funcional	El sistema debe de notificar a los padres de cualquier cambio en la evolución de sus hijos y de cualquier acontecimiento que involucre a su hijo.	Alta	Propuesto
80	Director de empresa	No Funcional	El sistema debe de permitir a la dirección comunicarse con los empleados ante cualquier percance o para comunicar alguna sugerencia.	Media	Propuesto
81	Director de empresa	No Funcional	El sistema solicitará de nuevo la clave del usuario cuando pase un determinado tiempo por haber inactividad y entender que ha caducado la sesión.	Alta	Propuesto
82	Director de empresa	No Funcional	La forma de loguearse en la aplicación debe de ser fácil para los empleados ya que no pueden perder mucho tiempo en ello.	Alta	Propuesto
83	Director de empresa	No Funcional	Los padres podrán notificar a la escuela infantil que hay un cambio en la recogida del hijo y para eso tendrán que indicar quien es la persona responsable.	Media	Propuesto
84	Director de empresa	No Funcional	Los padres podrán cambiar la dieta de su hijo mediante el sistema	Media	Propuesto
85	Director de empresa	No Funcional	Las comunicaciones entre los terminales y el servidor deben de ser seguras para que no haya accesos no deseados.	Media	Propuesto
86	Director de empresa	No Funcional	Los anuncios se enviarán a cada padre como notificación y además, las que sean de tipo general, se publicarán en el tablón de anuncios.	Media	Propuesto
ASI 2.2			<i>Especificación de Casos de Uso</i>		
87	Cocinero	No Funcional	Tiene que tener la posibilidad de variar el menú del día	Media	Propuesto
88	Padres	No Funcional	Tiene que tener la posibilidad de variar el menú del día	Media	Propuesto
89	Padres	No Funcional	Tiene que tener la posibilidad de cambiar la hora de recogida	Media	Propuesto

90	Padres	No Funcional	Tiene que tener la posibilidad de cambiar la persona que recoge al hijo	Media	Propuesto
91	Educadora	No Funcional	Facilidad de indicar la evolución educativa del alumno	Alta	Propuesto
92	Educadora	No Funcional	Facilidad de indicar incidencia o notas que ocurran con un alumno o acontecimiento	Alta	Propuesto
93	Educadora	No Funcional	Posibilidad de solicitar a los padres material para sus hijos	Media	Propuesto
94	Coordinadora	No Funcional	Avisar a los padres ante cualquier alerta de sus hijos	Alta	Propuesto
95	Administrativo	No Funcional	Facilidad para publicar notas en el tablón de anuncios	Media	Propuesto
96	Padres	No Funcional	Facilidad para visualizar los avances en incidencias de los hijos	Media	Propuesto
97	Educadora	No Funcional	Posibilidad de añadir materiales necesarios a la lista de productos para comprar	Media	Propuesto
98	Cocinero	No Funcional	Facilidad para realizar pedidos a distribuidores	Media	Propuesto
99	Administrativo	No Funcional	Centralización de la administración de dinero	Media	Propuesto
100	Coordinadora	No Funcional	Posibilitar el pago al contado de los padres que lo deseen	Media	Propuesto
101	Director de empresa	No Funcional	Cada empleado tiene que estar identificado cuando trabaje con el sistema	Alta	Propuesto
102	Director de empresa	No Funcional	Cada Padre tiene que estar identificado cuando opere con el sistema	Alta	Propuesto
103	Director de empresa	No Funcional	El sistema establece un tiempo de inactividad para desconectar a los usuarios del sistema. Y solo el director de la empresa tiene el privilegio de cambiar ese tiempo	Alta	Propuesto
104	Director de empresa	No Funcional	Facilidad de inicio de sesión para los empleados	Alta	Propuesto
105	Director de empresa	No Funcional	Todas las comunicaciones con el sistema de información deben de ir codificadas	Alta	Propuesto

ASI 8.1			<i>Especificación de Principios Generales de la Interfaz</i>		
106	Director de empresa	No Funcional	Interfaz Gráfico - Identificación de la empresa - Deben de aparecer los datos mínimos necesarios de la empresa, así como el logo. Además debe de haber una sección de contacto donde aparecerá la información completa.	Alta	Propuesto
107	Administrativo	No Funcional	Interfaz Gráfico - Identificación de la interfaz gráfica - Debe de aparecer el nombre de la sección en la que se encuentra el usuario.	Alta	Propuesto
108	Administrativo, Educadores y Padres	No Funcional	Interfaz Gráfico - Menú visible - Debe de estar siempre visible el menú del sistema correspondiente al rol del usuario.	Alta	Propuesto
109	Administrativo	No Funcional	Interfaz Gráfico - Identificación del usuario - Deben de aparecer siempre si es usuario anónimo o logado. Y si es logado debe de aparecer identificado siempre.	Alta	Propuesto
110	Administrativo, Educadores y Padres	No Funcional	Interfaz Gráfico - Uso de interfaz - El manejo de la interfaz por parte de los usuarios logados será del tipo “Draga & Drope”. Esto quiere decir que, dentro de las posibilidades de las tareas que tenga cada Rol de Usuario, se permitirá el manejo del sistema pulsando y arrastrando.	Alta	Propuesto
111	Administrativo y Educadores	No Funcional	Interfaz Gráfico - Tipo de letra - Los tipos de letras que se usarán serán el “Times Román” para las partes administrativas y de contabilidad de la empresa. Para el resto se usará el tipo “Comic”	Media	Propuesto
112	Administrativo y Educadores	No Funcional	Interfaz Gráfico - Uso de colores - Los colores que se usarán en todas las interfaces serán en tonalidades pastel.	Baja	Propuesto
113	Administrativo y Educadores	No Funcional	Interfaz Gráfico - Mostrar datos - La información se mostrará, siempre que se pueda, en una sola página. Si no pudiera ser así, y dependiendo de la sección del sistema en la que se encuentre el usuario, la información se mostrará en paginación o con barra de desplazamiento.	Media	Propuesto
114	Director de empresa	No Funcional	Impresos - Identificación de la empresa - Deben de aparecer los datos de la empresa así como el logo de la misma.	Alta	Propuesto
115	Administrativo	No Funcional	Impresos - Identificación del documento - Debe de aparecer el nombre del documento o hacer referencia al documento al que pertenece.	Alta	Propuesto

116	Administrativo	No Funcional	Impresos - Número de página - Debe de aparecer el número de página de cada documento así como el total de páginas al que pertenece.	Media	Propuesto
117	Administrativo	No Funcional	Impresos - Datos de información - Toda la información que muestra el documento debe de estar situado en el “Cuadro de Datos”. Este cuadro será el espacio del documento donde se imprimirá la información.	Media	Propuesto
118	Administrativo	No Funcional	Impresos - Tipos de letras - Para todos los documentos que se impriman se usará el tipo de letra “Time Roman”. Para los documentos que hagan referencia a Celebraciones o Fiestas se usará “Comic”.	Media	Propuesto
119	Administrativo y Educadores	No Funcional	Impresos - Uso de Colores - Todos los documentos se imprimirán en Blanco y Negro. Excepto los documentos que hagan referencia a Celebraciones o Fiestas que en tal caso si se usaran colores, pero de tonalidades pastel.	Baja	Propuesto

Tabla 20: Tabla de Requisitos

6.5 Apéndice V: Diagramas de casos de Uso.

En este capítulo se muestran los diagramas de casos de uso que se han realizado para representar la casuísticas, actores y acciones que permites desarrollar el proyecto y así cumplir con los requisitos planteados.

Diagrama de gestión de pedidos.

En este diagrama se establecen los diferentes actores que participaran en esta parte del sistema.

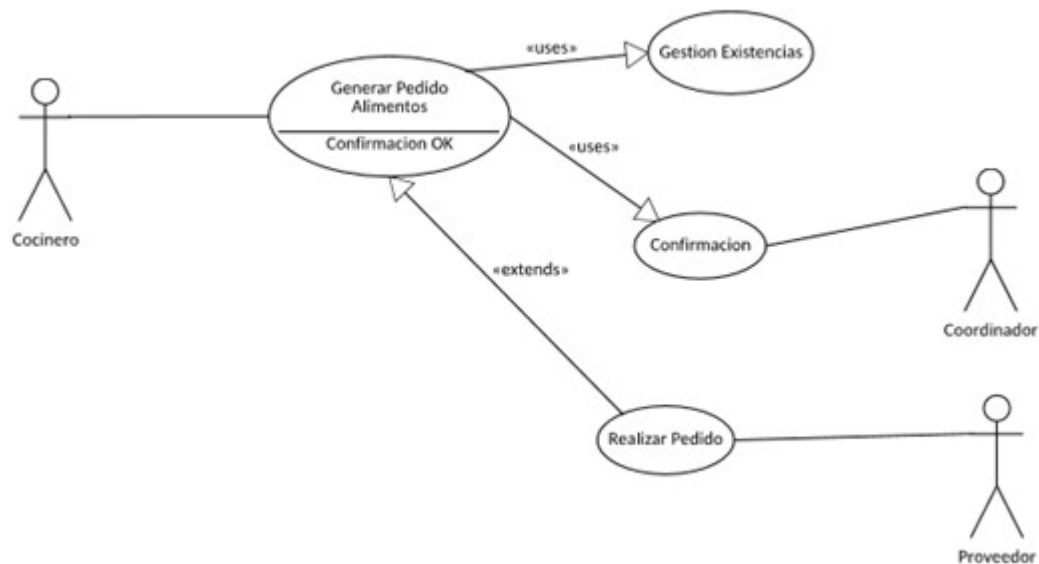


Figura 18: Caso de Uso de almacén 1

Diagrama de caso de uso de gestión de almacén en recepción de pedidos.

En este diagrama se muestran todas las posibles acciones que se pueden realizar cuando se reciben pedidos o materiales.

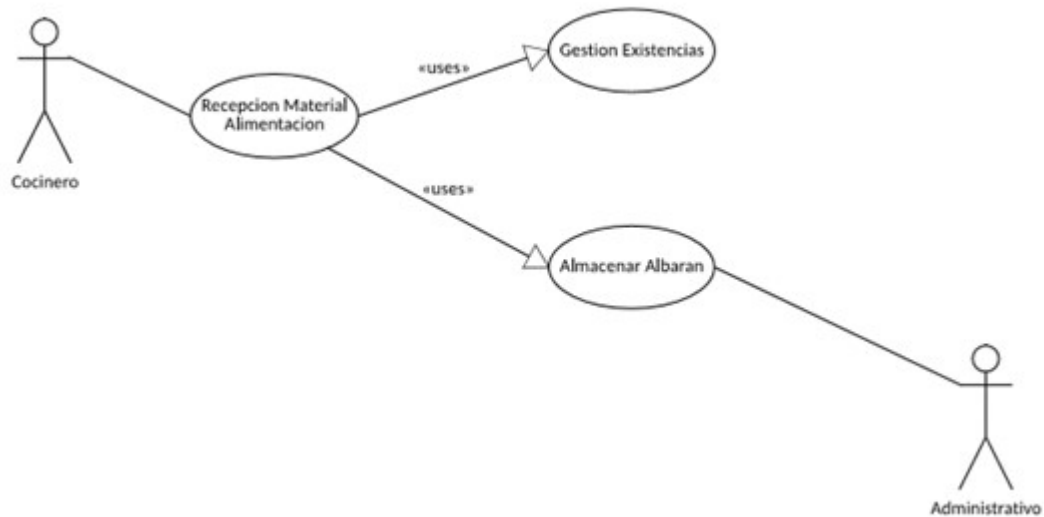


Figura 19: Caso de Uso de almacén 2

Diagrama de caso de uso de la gestión de comidas.

En este diagrama se muestran todas las posibles acciones que pueden realizar los cocineros, así como con los actores con los que interacciona.

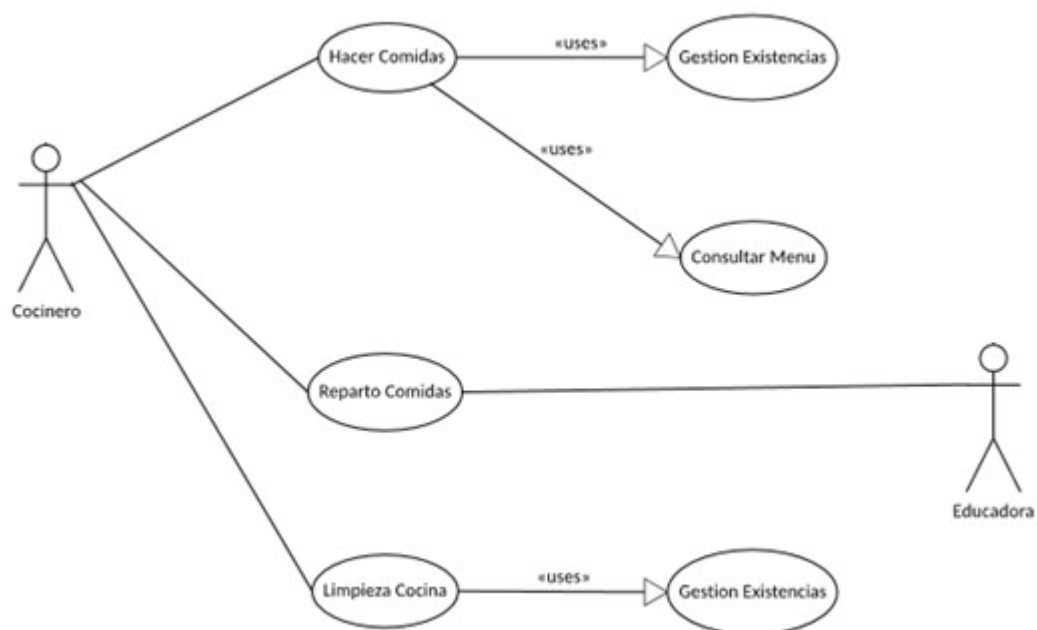


Figura 20: Caso de Uso de cocina

Diagrama de caso de uso de Gestión de Profesor

En este diagrama se muestran todas las posibles acciones que pueden realizar los educadores, así como con los actores con los que interacciona.

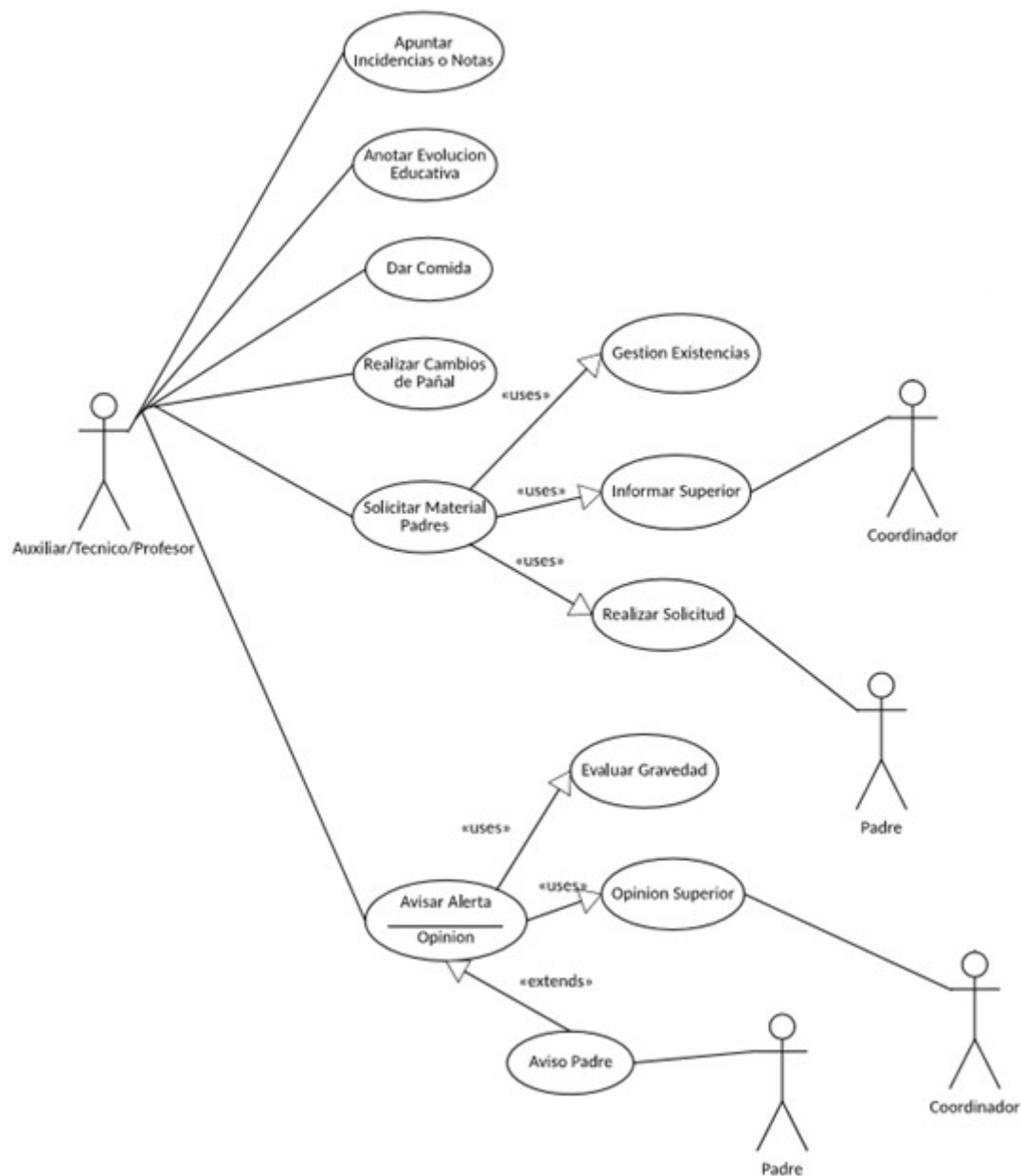


Figura 21: Caso de Uso de Profes

Diagrama de caso de uso de Gestión de Padres.

En este diagrama se muestran todas las posibles acciones que pueden realizar los padres, así como con los actores con los que interacciona.

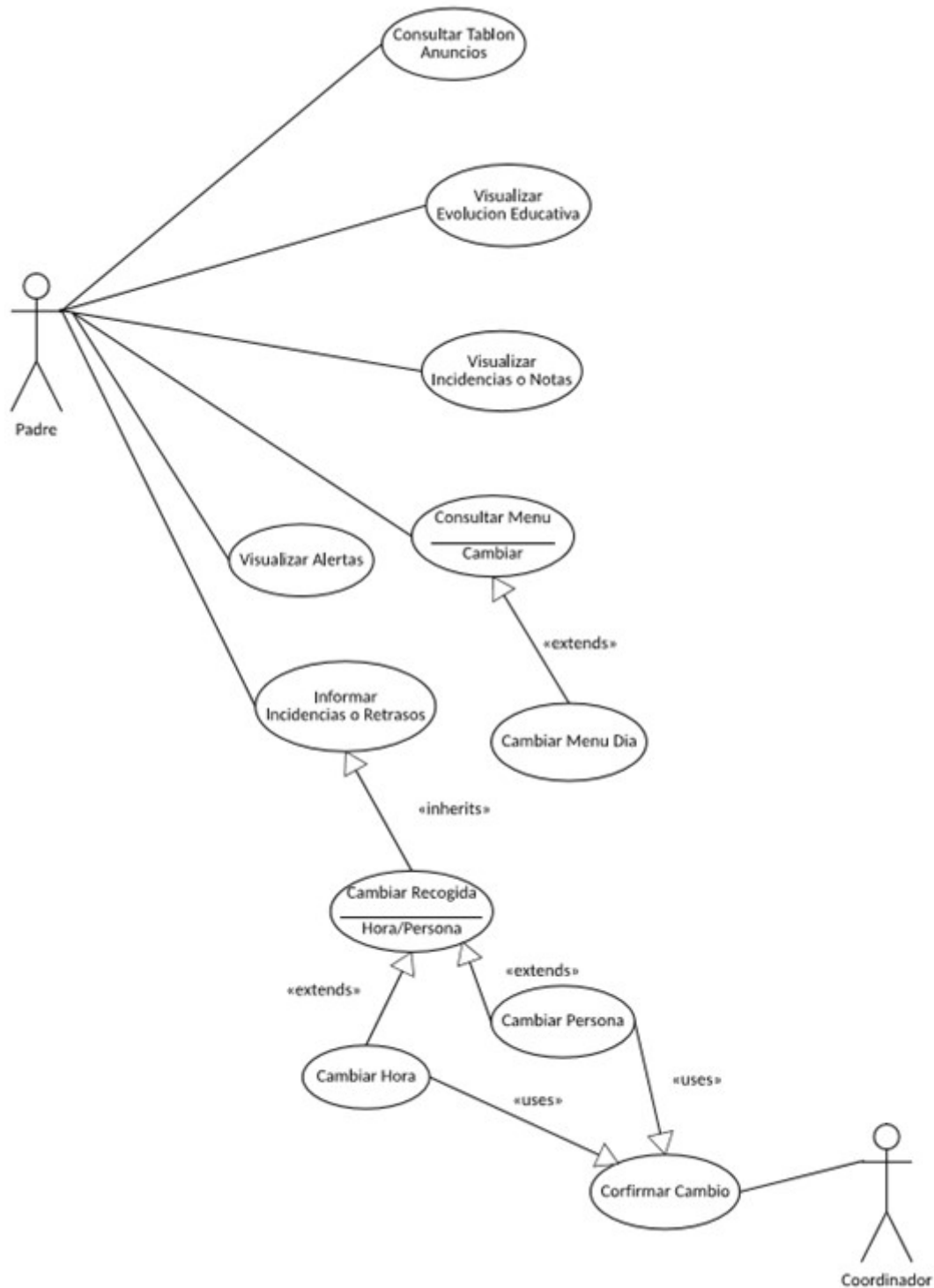


Figura 22: Caso de Uso de Padres

6.6 Apéndice VI: Escenarios de Casos de Uso.

Aquí se pueden mostrar una tabla en la que figuran los escenarios básicos de cada uno de los casos de uso que se han mostrado en los diagramas del Apéndice IV.

Escenario Normal

Escenarios de Casos de Uso	Generar Pedido (Alimentos)
Pasos	Descripción
1	El cocinero gestiona las existencias
2	Establece el listado de productos
3	Envía listado a superior
4	Recibe confirmación de listado
5	Envía listado a los proveedores

Escenarios de Casos de Uso	Gestionar Existencias (Alimentos)
Pasos	Descripción
1	Visualiza cantidades de productos
2	Evaluar estado productos perecederos
3	Evaluar estado productos no perecederos
4	Evaluar necesidades mínimas
5	Generar listado de productos necesarios

Escenario de Excepción

Pasos	Descripción
1	El cocinero gestiona las existencias
2	Establece el listado de productos
3	Envía listado a superior
4	Recibe negativa de superior y las razones

Pasos	Descripción

Escenarios de Casos de Uso	Confirmar
Pasos	Descripción
1	Recibir listado de productos necesarios
2	Evaluar listado con el presupuesto indicado
3	Enviar a administrativo cantidades en previsión de gastos y futuras entradas
4	Confirmar listado y presupuesto

Escenarios de Casos de Uso	Realizar Pedido
Pasos	Descripción
1	Realizar pedido a los diferentes proveedores
2	Anotar listado de productos y cantidades en previsión de entrada

Escenarios de Casos de Uso	Recepción de Materiales (Alimentación y Limpieza)
Pasos	Descripción
1	El cocinero recibe los productos de los diferentes proveedores
2	Evalúa estado y cantidades
3	Confirma entrada y gestiona las existencias
4	Envía albarán a administrativo para contabilidad

Escenarios de Casos de Uso	Almacenar Albarán
Pasos	Descripción
1	El administrativo recibe albarán
2	Anota los productos, cantidades y precios

Pasos	Descripción
1	Recibir listado de productos necesarios
2	Evaluar listado con el presupuesto indicado
3	Denegar listado e indicar razones

Pasos	Descripción

Pasos	Descripción
1	El cocinero recibe los productos de los diferentes proveedores
2	Evalúa estado y cantidades
3	Rechaza productos y pone las razones

Pasos	Descripción

Escenarios de Casos de Uso	Hacer Comidas
Pasos	Descripción
1	El cocinero consulta el menú del día
2	Gestiona existencias
3	Hace comidas establecidas

Escenarios de Casos de Uso	Consultar Menú
Pasos	Descripción
1	Consulta menú del día programado
2	Consulta menús de excepción
3	Elige alternativas

Escenarios de Casos de Uso	Reparto Comidas
Pasos	Descripción
1	El cocinero organiza los menús
2	Indica menú especiales
3	Avisa a las respectivas aulas
4	Los educadores recogen la comida de sus respectivas clases

Escenarios de Casos de Uso	Limpieza Cocina
Pasos	Descripción
1	El cocinero recoge todos los utensilios de cocina
2	Gestiona existencias de productos de limpieza
3	Limpia la cocina

Pasos	Descripción

Pasos	Descripción

Pasos	Descripción

Pasos	Descripción
1	El cocinero recoge todos los utensilios de cocina
2	Gestiona existencias de productos de limpieza
3	Si faltan productos lo indica
4	Limpia cocina si puede

Escenarios de Casos de Uso	Limpieza Aulas
Pasos	Descripción
1	Los educadores recogen los artículos que haya repartidos por sus respectivas clases
2	Gestiona existencias de productos de limpieza
3	Limpia el aula

Escenarios de Casos de Uso	Generar Pedido (Material)
Pasos	Descripción
1	Empleado gestiona existencias de productos
2	Superior revisa listado de pedido
3	Confirma listado y generar listado
4	Realizar pedido a proveedor

Escenarios de Casos de Uso	Recepción de Materiales (Materiales)
Pasos	Descripción
1	El coordinador recibe los materiales
2	Revisa listado
3	Añade o administra existencias
4	Envía albarán a contabilidad

Pasos	Descripción
1	Los educadores recogen los artículos que haya repartidos por sus respectivas clases
2	Gestiona existencias de productos de limpieza
3	Si faltan productos lo indica
4	Limpia aula si puede

Pasos	Descripción
1	Empleado gestiona existencias de productos
2	Superior revisa listado de pedido
3	Cancela o modifica listado y pone razón

Pasos	Descripción

Escenarios de Casos de Uso	Realizar Pagos
Pasos	Descripción
1	Administrativo revisa cuentas del mes
2	Administrativo revisa albaranes
3	Realiza pagos de nominas
4	Realiza pagos de albaranes

Escenarios de Casos de Uso	Ingreso Nominas
Pasos	Descripción
1	Establecer empleado
2	Establecer cantidades a ingresar
3	Establecer número de cuenta
4	Ordenar a banco el ingreso en número de cuenta

Escenarios de Casos de Uso	Ingreso Albarán
Pasos	Descripción
1	Establecer proveedor
2	Establecer cantidades a ingresar
3	Establecer número de cuenta
4	Ordenar a banco el ingreso en número de cuenta

Pasos	Descripción

Pasos	Descripción

Pasos	Descripción
1	Establecer proveedor
2	Establecer cantidades a ingresar
3	Contactar con proveedor y establecer día de pago
4	Pagar cantidad
5	Firmar cobro de cantidad por proveedor

Escenarios de Casos de Uso	Realizar Cobros
Pasos	Descripción
1	Administrativo revisa cuentas del mes
2	Realiza cobros de administraciones
3	Realiza cobros de domiciliaciones
4	Envía aviso de cobros en caja a coordinador
5	Coordinador realiza cobro mensualidad en caja

Escenarios de Casos de Uso	Cobro Administraciones Publicas
Pasos	Descripción
1	Realizar listado de cantidades a cobrar a administración
2	Indicar cuenta bancaria donde ingresar
3	Envío de listado a administración

Escenarios de Casos de Uso	Cobros Domiciliación
Pasos	Descripción
1	Establecer cantidades a cobrar
2	Indicar razón
3	Indicar cuenta
4	Ordenar a banco el cobro

Pasos	Descripción

Pasos	Descripción

Pasos	Descripción

Escenarios de Casos de Uso	Cobro en Caja
Pasos	Descripción
1	Establecer cantidades a cobrar
2	Indicar razón de cada cantidad
3	Contar dinero
4	Firmar la recaudación del dinero

Escenarios de Casos de Uso	Apuntar Incidencia o Notas
Pasos	Descripción
1	Seleccionar si es genérica o pertenece a un alumno
2	Anotar la incidencia, o nota, y la hora en la que ocurrió
3	Breve explicación de la incidencia

Escenarios de Casos de Uso	Anotar Evolución Educativa
Pasos	Descripción
1	Seleccionar al alumno
2	Rellenar el cuestionario

Pasos	Descripción
1	Establecer cantidades a cobrar
2	Indicar razón de cada cantidad
3	Contar dinero
4	Rechazar el cobro e indicar la razón

Pasos	Descripción

Pasos	Descripción

Escenarios de Casos de Uso	Dar Comida
Pasos	Descripción
1	Recibir los alimentos de la cocina
2	Identificar los platos especiales para cada alumno
3	Dar de comer
4	Recoger los platos y cubiertos
5	Dejarlos en cocina

Escenarios de Casos de Uso	Realizar Cambios de Pañal
Pasos	Descripción
1	Identificar alumno que necesita cambio
2	Gestionar existencias pertenecientes al alumno
3	Realizar cambio de pañal

Escenarios de Casos de Uso	Solicitar Material Padres
Pasos	Descripción
1	Gestionar existencias de los alumnos
2	Generar listado de materiales por alumno
3	Informar a superior
4	Solicitar material a padres

Pasos	Descripción

Pasos	Descripción

Pasos	Descripción

Escenarios de Casos de Uso	Informar Superior
Pasos	Descripción
1	Enviar listado de materiales de alumnos asociado a padres
2	Establecer como notificación de aviso en cuenta de superior

Escenarios de Casos de Uso	Realizar Solicitud
Pasos	Descripción
1	Enviar listado de materiales de alumnos a sus respectivos padres

Escenarios de Casos de Uso	Avisar Alerta
Pasos	Descripción
1	Evaluar la gravedad de la alerta
2	Informar a superior y esperar opinión
3	Seguir protocolos o medidas convenientes
4	Redactar informar de acontecimientos

Escenarios de Casos de Uso	Evaluar Gravedad
Pasos	Descripción
1	Visualizar los acontecimientos o hechos
2	Visualizar consecuencias
3	Establecer gravedad de consecuencias

Pasos	Descripción

Pasos	Descripción

Pasos	Descripción
1	Evaluar la gravedad de la alerta
2	Informar a superior y esperar opinión
3	Seguir protocolos o medidas convenientes
4	Avisar a padres
5	Seguir protocolos o medidas convenientes
6	Redactar informar de acontecimientos

Pasos	Descripción

Escenarios de Casos de Uso	Opinión Superior
Pasos	Descripción
1	Avisar a superior de los acontecimientos
2	Informar de la gravedad de la situación
3	Informar de las medidas tomadas
4	Esperar opinión y razones

Escenarios de Casos de Uso	Aviso Padre
Pasos	Descripción
1	Buscar alumnos afectados en sistema
2	Indicar situación
3	Enviar mensaje de alerta a los padres

Escenarios de Casos de Uso	Publicar Tablón Anuncios
Pasos	Descripción
1	Administrativo establece nota genérica a publicar
2	Indica hora de la nota
3	Indica título de la nota
4	Desarrolla nota
5	Publica nota

Pasos	Descripción

Pasos	Descripción
1	Buscar alumnos afectados en sistema
2	Indicar situación
3	Marcar que padres contacten con empresa
4	Enviar mensaje de alerta a los padres

Pasos	Descripción

Escenarios de Casos de Uso	Publicar Tablón Anuncios Local
Pasos	Descripción
1	Coordinador establece nota local a publicar
2	Indica hora de la nota
3	Indica título de la nota
4	Desarrolla nota
5	Publica nota

Escenarios de Casos de Uso	Consultar Tablón Anuncios
Pasos	Descripción
1	Visualizar notas genérica publicadas

Escenarios de Casos de Uso	Visualizar Evolución Educativa
Pasos	Descripción
1	Visualizar evolución de sus hijos

Escenarios de Casos de Uso	Visualizar Incidencias o Notas
Pasos	Descripción
1	Visualizar notas o incidencias de sus hijos

Escenarios de Casos de Uso	Consultar Menú Día
Pasos	Descripción
1	Visualizar programa del menú de sus hijos

Pasos	Descripción

Pasos	Descripción
1	Visualizar notas locales publicadas

Pasos	Descripción

Pasos	Descripción

Pasos	Descripción
1	Visualizar programa del menú de sus hijos
2	Visualizar alternativas

Escenarios de Casos de Uso	Cambiar Menú Día
Pasos	Descripción
1	Consultar menú
2	Visualizar alternativas para sus hijos
3	Ver hora límite de cambio
4	Cambiar menú por alternativa seleccionada

Escenarios de Casos de Uso	Visualizar Alertas
Pasos	Descripción
1	Visualizar alertas de sus hijos

Escenarios de Casos de Uso	Informar Incidencias o Retrasos
Pasos	Descripción
1	Indicar incidencia o retraso
2	Indicar título de incidencia
3	Desarrollar indecencia

Escenarios de Casos de Uso	Cambiar Recogida
Pasos	Descripción
1	Indicar cambio de recogida
2	Indicar cambio de hora

Pasos	Descripción
1	Consultar menú
2	Visualizar alternativas para sus hijos
3	Sobrepasada hora limite

Pasos	Descripción
1	Visualizar alertas de sus hijos
2	Contactar con centro

Pasos	Descripción

Pasos	Descripción
1	Indicar cambio de recogida
2	Indicar cambio de persona

Escenarios de Casos de Uso	Cambiar Hora
Pasos	Descripción
1	Seleccionar nueva hora de recogida
2	Enviar solicitud
3	Esperar confirmación o razones de rechazo

Escenarios de Casos de Uso	Cambiar Persona
Pasos	Descripción
1	Seleccionar nueva persona de recogida
2	Seleccionar nueva hora de recogida
3	Enviar solicitud
4	Esperar confirmación o razones de rechazo

Escenarios de Casos de Uso	Confirmar Cambio
Pasos	Descripción
1	Coordinador recibe nota de cambio de hora o persona de recogida
2	Evalúa nueva hora de recogida
3	Envía confirmación

Pasos	Descripción

Pasos	Descripción

Pasos	Descripción
1	Coordinador recibe nota de cambio de hora o persona de recogida
2	Busca y evalúa datos de nueva persona
3	Evalúa nueva hora de recogida
4	Envía confirmación

Escenarios de Casos de Uso	Recepción Material Padres
Pasos	Descripción
1	Coordinador recibe material de alumno por parte de padre
2	Visualiza listados de solicitudes de materiales
3	Marca como recibido
4	Suministra el material a la educadora del alumno

Escenarios de Casos de Uso	Suministrar Material Aula
Pasos	Descripción
1	Educadora recibe material del alumno
2	Gestiona existencias

Escenarios de Casos de Uso	Navegar por Sistema
Pasos	Descripción
1	El usuario selecciona las opciones del menú
2	El sistema gestiona sesión
3	El sistema recupera la información necesaria
4	El sistema muestra la información solicitada al usuario

Pasos	Descripción

Pasos	Descripción

Pasos	Descripción
1	El usuario selecciona las opciones del menú
2	El sistema gestiona sesión
3	Deniega permiso
4	Muestra mensaje de información restringida

Escenarios de Casos de Uso	Navegar por Sistema Especifico
Pasos	Descripción
1	El usuario selecciona las opciones específicas del menú
2	El sistema gestiona sesión
3	El sistema recupera la información específica
4	El sistema muestra la información solicitada

Escenarios de Casos de Uso	Gestión de Sesión
Pasos	Descripción
1	El sistema comprueba identidad de usuario
2	Comprueba permisos de usuario

Escenarios de Casos de Uso	Gestión de Recursos
Pasos	Descripción
1	El empleado accede con su usuario
2	Gestión de sesión
3	Selecciona el recurso que quiere administrar
4	Da de alta, actualiza o elimina el recurso

Escenarios de Casos de Uso	Gestión de Empleados
Pasos	Descripción
1	El empleado accede con su usuario
2	Gestión de sesión
3	Da de alta, actualiza o elimina a empleados

Pasos	Descripción
1	El usuario selecciona las opciones específicas del menú
2	El sistema gestiona sesión
3	Deniega solicitud
4	Muestra mensaje de información restringida

Pasos	Descripción
1	

Pasos	Descripción
1	El empleado accede con su usuario
2	Gestión de sesión
3	Acceso denegado

Pasos	Descripción
1	El empleado accede con su usuario
2	Gestión de sesión
3	Acceso denegado

Escenarios de Casos de Uso	Gestión de Alumnos
Pasos	Descripción
1	El empleado accede con su usuario
2	Gestión de sesión
3	Da de alta, actualiza o elimina a alumnos

Escenarios de Casos de Uso	Gestión de Padres o Personas de Interés
Pasos	Descripción
1	El empleado accede con su usuario
2	Gestión de sesión
3	Da de alta, actualiza o elimina a padres o personas de interés para los alumnos

Escenarios de Casos de Uso	Gestión de Materiales
Pasos	Descripción
1	El empleado accede con su usuario
2	Gestión de sesión
3	Da de alta, actualiza o elimina a materiales

Pasos	Descripción
1	El empleado accede con su usuario
2	Gestión de sesión
3	Acceso denegado

Pasos	Descripción
1	El empleado accede con su usuario
2	Gestión de sesión
3	Acceso denegado

Pasos	Descripción
1	El empleado accede con su usuario
2	Gestión de sesión
3	Acceso denegado

Tabla 21: Tabla de Escenarios de Casos de Uso

6.7 Apéndice VII: Sitio Web

En este apéndice se muestra el estado de algunas pantallas del sitio Web tras el diseño y la implementación del mismo.

Pantalla de inicio o “Home”.

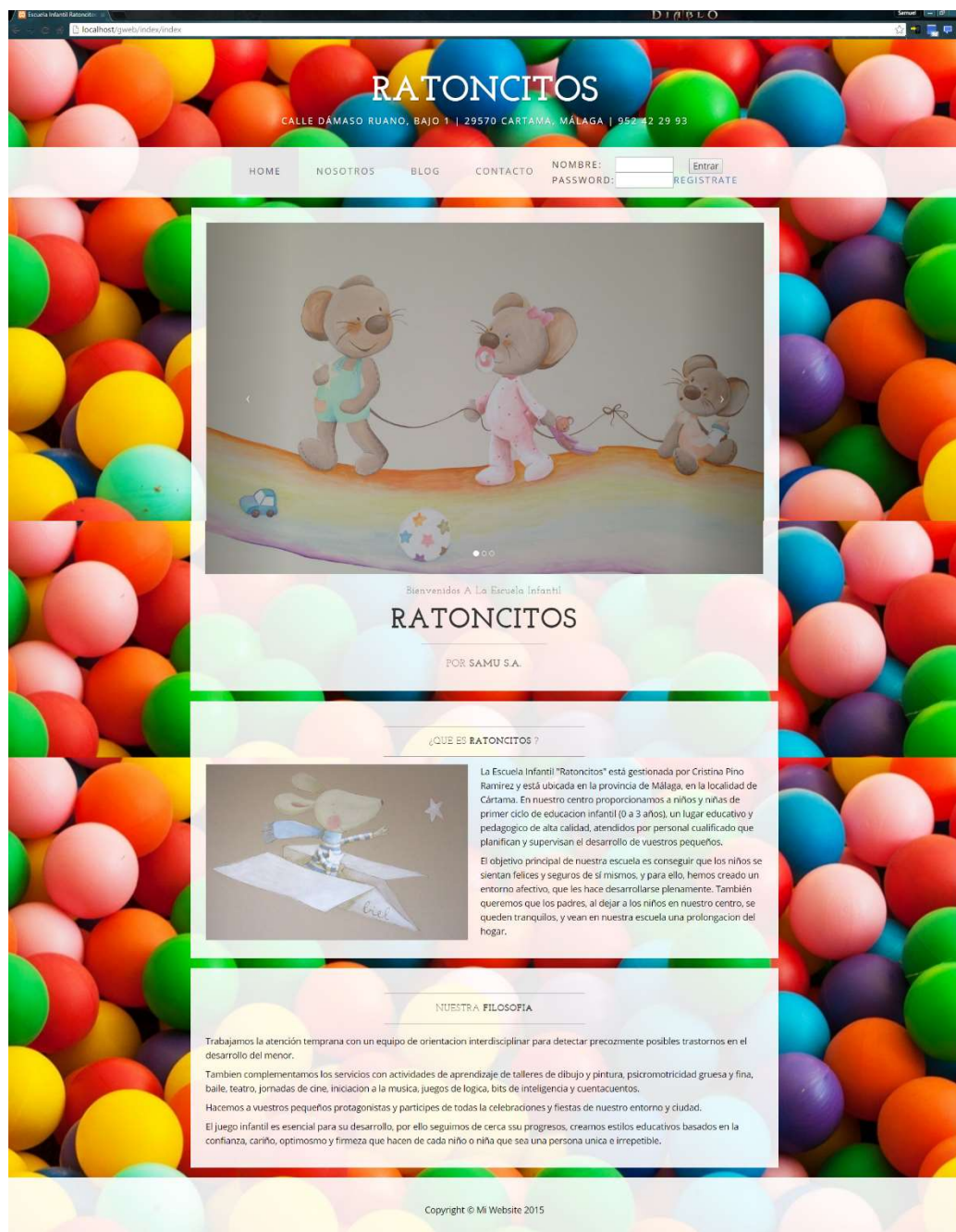


Figura 23: Pagina Home

Página Nosotros.



Figura 24: Pagina Nosotros

Página Blog.

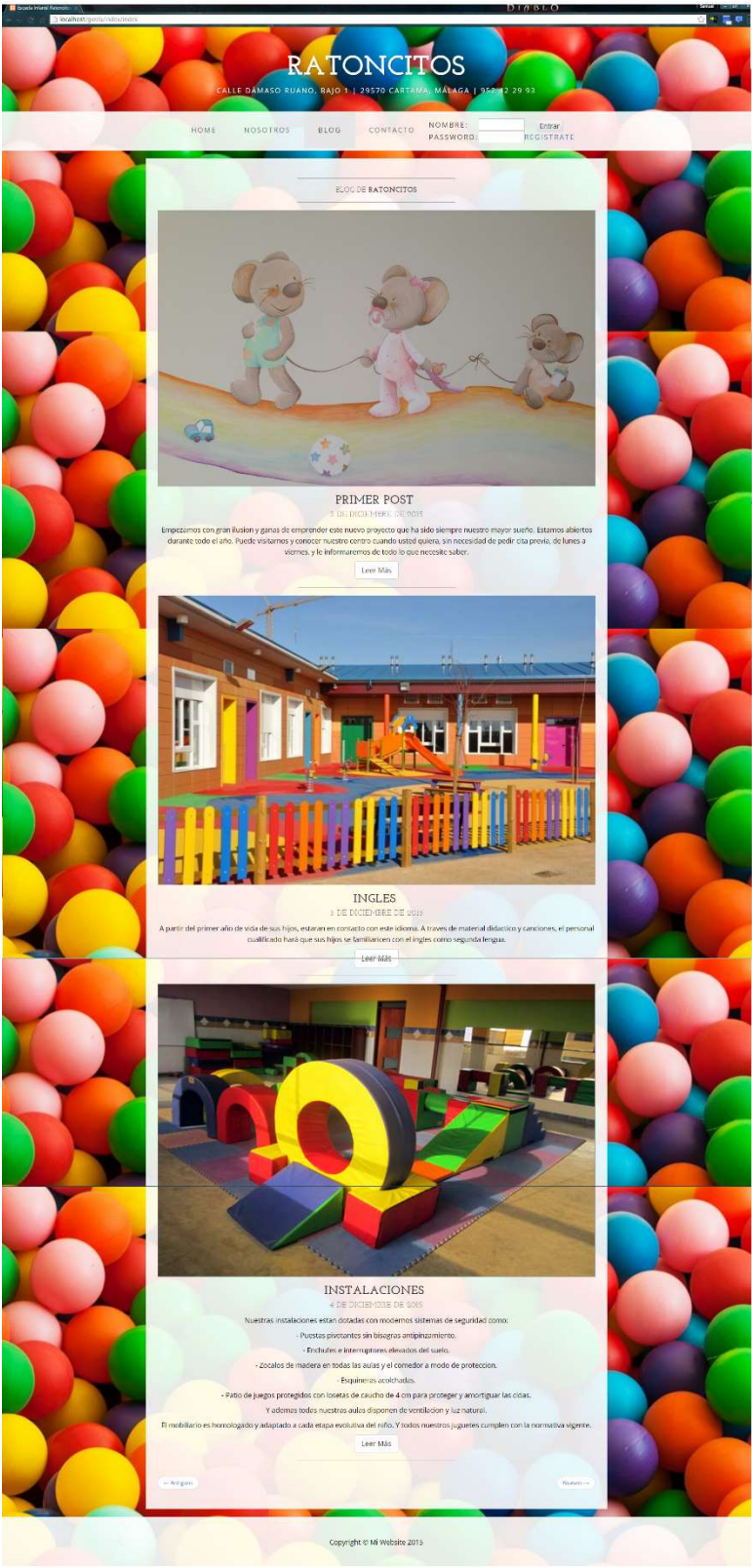


Figura 25: Pagina Blog

Página Contacto.

Escuela Infantil Ratoncitos

localhost/gweb/index/index

RATONCITOS
CALLE DÁMASO RUANO, BAJO 1 | 29570 CARTAMA, MÁLAGA | 952 42 29 93

HOME NOSOTROS BLOG CONTACTO

NOMBRE: Entrar
PASSWORD: REGISTRATE

SITUACION DE RATONCITOS

View larger map

Telefono: 952 42 29 93
Email: samu@samu.com
Dirección: Calle Dámaso Ruano, bajo 1
29570 Cartama, Málaga
Horario:
Lunes a Viernes : 7:15 - 19:00
Aula Matinal : 7:15 - 9:00
Ludoteca : 17:00 - 19:00

FORMULARIO DE CONTACTO

Para contactar con nosotros, por favor rellene los siguientes campos y nos pondremos en contacto con usted lo antes posible.

Nombre Dirección de Email Número de Teléfono

Mensaje

Enviar

Copyright © Mi Website 2015

Figura 26: Página Contacto

Página usuario logueado y en su menú.

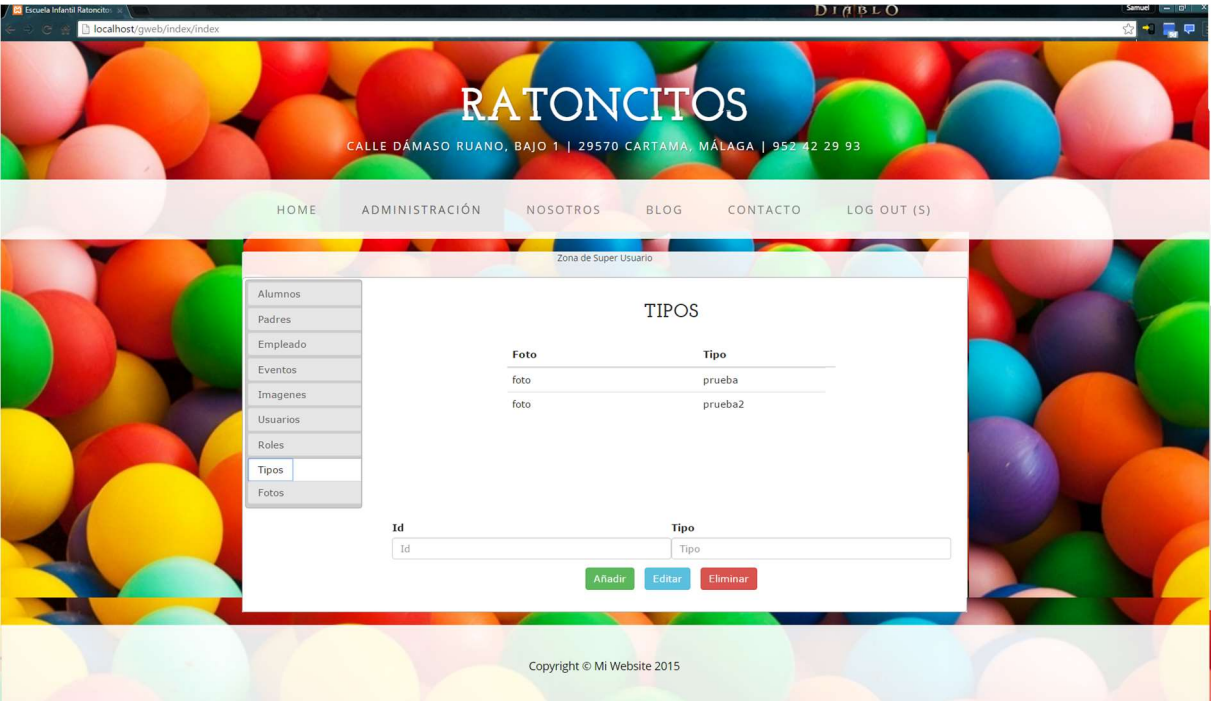


Figura 27: Pagina usuario logueado y en su menú

Página Home Responsive



Figura 28: Pagina Home Responsive

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
- [3] Web oficial de PhalconPHP. <https://www.phalconphp.com/es/>
- [4] Página web oficial de PHP. 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>
- [13] Página Web oficial de Métrica v3.
http://administracionelectronica.gob.es/pae_Home/pae_Documentacion/pae_Metodolog/pae_Metrica_v3.html#.Vm1ZivnhCHs
- [14] Página Web de “*Qué es SCRUM*”. <http://proyectosagiles.org/que-es-scrum>