

UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA
ESCUELA DE INGENIERÍA DE
TELECOMUNICACIÓN Y ELECTRÓNICA



PROYECTO FIN DE CARRERA

DESARROLLO DE UNA APLICACIÓN PARA MOVILES
PARA LA DIFUSION DE EVENTOS CULTURALES
POR ORGANISMOS OFICIALES

TITULACIÓN: INGENIERO TÉCNICO DE TELECOMUNICACIÓN
EN TELEMÁTICA

TUTORES: JOSÉ MARÍA QUINTEIRO GONZÁLEZ
DAVID DE LA CRUZ SÁNCHEZ RODRÍGUEZ

AUTOR: RUYMÁN RODRÍGUEZ GONZÁLEZ

FECHA: DICIEMBRE 2014

UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA
ESCUELA DE INGENIERÍA DE
TELECOMUNICACIÓN Y ELECTRÓNICA



PROYECTO FIN DE CARRERA

DESARROLLO DE UNA APLICACIÓN PARA MOVILES PARA LA DIFUSION
DE EVENTOS CULTURALES POR ORGANISMOS OFICIALES

Presidente:

Secretario:

Vocal:

Tutores:

Autor:

NOTA:

TITULACIÓN: INGENIERO TÉCNICO DE TELECOMUNICACIÓN
EN TELEMÁTICA

TUTORES: JOSÉ MARÍA QUINTEIRO GONZÁLEZ
DAVID DE LA CRUZ SÁNCHEZ RODRÍGUEZ

AUTOR: RUYMÁN RODRÍGUEZ GONZÁLEZ

FECHA: DICIEMBRE 2014

Bloque 1º. Memoria.

Capítulo 1: Introducción	6
1.1 Introducción.....	6
1.2 Objetivos.....	6
1.3 Estructura de la memoria.....	7
Capítulo 2: Arquitectura del proyecto	9
2.1 Gestor de contenidos.....	11
2.1.1 Distintos CMS del mercado.....	12
2.2 Creación de formularios para el portal web.....	16
2.3 Computación en la nube.....	18
2.3.1 Servicios en la nube utilizados.....	24
Capítulo 3: Plataformas de desarrollo	31
3.1 WordPress.....	31
3.1.1 Administración de WordPress.....	34
3.1.2 Child Themes.....	40
3.1.3 Meta box.....	43
3.1.4 SEO.....	46
3.1.5 Configuración de una página o entrada.....	49
3.1.6 Creación de entradas y páginas.....	51
3.2 Zend Framework 2.....	55
3.2.1 MVC.....	55
3.2.2 Framework.....	58
3.2.3 MVC en aplicaciones web.....	59
3.2.4 Zend Framework 2.....	60
3.2.5 Zend Framework 2 en el sistema.....	62
Capítulo 4: Servicios en la nube	64
4.1 Heroku.....	64
4.1.1 Dynos.....	66
4.1.2 Modificación de la aplicación en Heroku.....	66
4.2 CDN.....	67
4.2.1 Amazon Web Service.....	69
4.3 Parse.....	74

4.3.1 Trabajando con Parse.....	76
Capítulo 5: Diseño y estructura del portal web	78
5.1 Diseño web.....	78
5.1.1 Estructura de una página web.....	79
5.1.2 Web adaptativa.....	81
5.1.3 Diseño inicial del portal web.....	83
5.1.4 Estructura de la web.....	88
5.1.5 Plantilla de WordPress.....	90
5.1.6 Creación del logotipo.....	92
5.2 Comprobación de la web en varios navegadores.....	93
5.2.1 Estudio del uso de navegadores.....	93
5.2.2 Comprobación de compatibilidad del portal web.....	96
5.2.3 Navegadores para dispositivos móviles.....	98
Capítulo 6: Diseño y estructura de la aplicación	103
6.1 SpiderCatalog.....	103
6.1.1 Estructura de SpiderCatalog.....	106
6.2 Estructura del sistema.....	108
6.2.1 Filtros.....	110
6.3 Adaptación del sistema a la estructura de SpiderCatalog.....	115
6.3.1 Configuraciones de la app.....	117
6.4 Ejemplos.....	119
Capítulo 7: Conclusiones	126
Capítulo 8: Bibliografía	127
 Bloque 2º. Pliego de condiciones.	
Capítulo 1: Pliego de condiciones.....	131
1.1 Lenguajes de programación.....	131
1.1.1 HTML.....	131
1.1.2 CSS.....	133
1.1.3 Javascript.....	134
1.1.4 PHP.....	135

1.1.5 JSON.....	136
1.2 Herramientas utilizadas.....	137
1.2.1 XAMPP.....	137
1.2.2 PHPStorm.....	138
1.2.3 Balsamiq mockup.....	138
1.2.4 GIMP.....	138
1.2.5 GIT.....	139
1.3 Instalación de servicios.....	140
1.3.1 Instalación de un servidor local.....	140
1.3.2 Instalación de WordPress.....	144
1.3.3 Instalación de ZF2.....	148
1.3.4 Instalación de WordPress en Heroku.....	150
Capítulo 2: Pliego de condiciones legales.....	153
2.1 Concesión de licencia.....	153
2.2 Derechos de autor.....	153
2.3 Restricciones.....	153
2.4 Limitación de responsabilidad.....	154
2.5 Varios.....	154
Capítulo 3: Guías de usuario.....	155
3.1 Guía de usuario web.....	155
3.2 Guía de usuario de la aplicación.....	161
Bloque 3º. Presupuesto.	
Capítulo 1: Presupuesto.....	171
1.1 Tiempo empleado en la realización del PFC.....	171
1.2 Amortización de las herramientas de hardware y software.....	173
1.3 Presupuesto total del Proyecto Fin de Carrera.....	174

Memoria

Introducción

1.1 Introducción

El proyecto fin de carrera *Desarrollo de una aplicación para móviles para la difusión de eventos culturales por organismos oficiales* tiene como propósito la creación de un espacio en el que informar a los habitantes de la isla de Gran Canaria de las actividades culturales que ofrecen los distintos organismos de la isla. Es de vital importancia la cultura en las vidas de las personas por lo que es necesario crear un medio adaptado a las nuevas tecnologías que funcione de altavoz a todos estos eventos.

La realización de la aplicación tendrá como consecuencia la creación de un portal web que tendrá dos funciones principales:

- Ser la fuente de contenidos de la que se nutra la aplicación.
- Otra vía de comunicación para el usuario final donde estar informado de las actividades culturales de Gran Canaria.

1.2 Objetivos

Los objetivos perseguidos y que han motivado la realización de este Proyecto Fin de Carrera han sido los siguientes:

- El aprendizaje de nuevos lenguajes de programación enfocados a la creación y diseño de portales web y aplicaciones.
- El estudio y uso de los gestores de contenidos que ofrece el mercado para la creación de portales web.
- Iniciarse en el patrón modelo-vista-controlador.
- Introducirse en el mundo de los servicios en la nube. Conocer cuáles son sus características para entender un nuevo método de trabajar para la creación de portales web.

1.3 Estructura de la memoria

El presente documento se encuentra dividido en tres partes principales: memoria, pliego de condiciones, presupuesto.

La memoria está estructurada en ocho capítulos.

- En el capítulo 2 se explica con detalle la arquitectura completa que conforma el sistema.
- En el capítulo 3 se presentan las plataformas de desarrollo utilizadas para surtir de contenidos al sistema e interacción con los usuarios vía web.
- En el capítulo 4 se presentan los servicios en la nube utilizados y la función que realizan cada uno de ellos en el sistema.
- En los capítulos 5 y 6 se muestran el diseño y la estructura del portal web y de la aplicación.
- En el capítulo 7 se incluyen las conclusiones obtenidas al realizar el proyecto.
- Al final de la memoria, en el capítulo 8, se detalla la bibliografía consultada durante la realización del proyecto.

El segundo bloque está constituido por tres capítulos que componen el pliego de condiciones:

- El pliego de condiciones donde se detallan los lenguajes de programación y los programas utilizados para la elaboración del proyecto.
- El pliego de condiciones legales que detalla los derechos de autor del Proyecto Fin de Carrera.
- Las guías de usuario del portal web y de la aplicación móvil.

El tercer bloque del documento lo conforma el presupuesto del proyecto,

que señala las fases en las que se divide la realización del proyecto y los costes necesarios de cada una de estas fases.

Arquitectura desarrollada en este PFC

La decisión tomada en este Proyecto Fin de Carrera para realizar la creación de una aplicación web que sirva como fuente de contenidos a una aplicación móvil implica dos grandes decisiones a tomar a la hora de crear la arquitectura del sistema:

- El estudio de las plataformas de desarrollo que crearán el contenido e interactuarán con el usuario final en el portal web.
- El estudio para la utilización de servicios en la nube para realizar el alojamiento web que dará soporte al portal web y el tratamiento de la información generada por el sistema para presentarla en el portal web y en la aplicación móvil.

Estas dos tareas principales a realizar se pueden dividir en cuatro bloques claramente diferenciados como se expone en la figura 2.1 y que se procede a explicar a continuación.

- **Bloque de contenidos.** Para crear y manejar los contenidos que incluidos en la web se opta por la utilización de un gestor de contenidos web (WCM). Se ha escogido la utilización de un WCM frente al desarrollo propio de una web desde cero para reducir el tiempo empleado en el desarrollo de diseño y desarrollo propio de la web. La utilización de gestores de contenidos de código abierto permiten a los desarrolladores gran libertad para la modificación y creación de código propio, por lo que permiten crear portales web adaptados a la necesidad del creador.

La utilización de uno de los gestores de contenidos que ofrece el mercado implica la realización de un estudio para determinar qué gestor de contenidos es el que más se adapta a las necesidades del proyecto. Una vez elegido el gestor de contenidos se procede a la familiarización con el entorno elegido para la creación y administración del portal web.

- **Bloque de formularios de entrada.** Este bloque está compuesto por unos formularios que permiten a los usuarios interactuar con el sistema. Se crearán dos tipos de formularios; uno para que los usuarios puedan darse de alta en el sistema y otro para que los usuarios tengan la opción de comunicarse con los administradores ante cualquier incidencia, petición o duda.

Dichos formularios se realizan con la utilización de un *framework* (marco de trabajo) bajo el patrón MVC (modelo-vista-controlador).

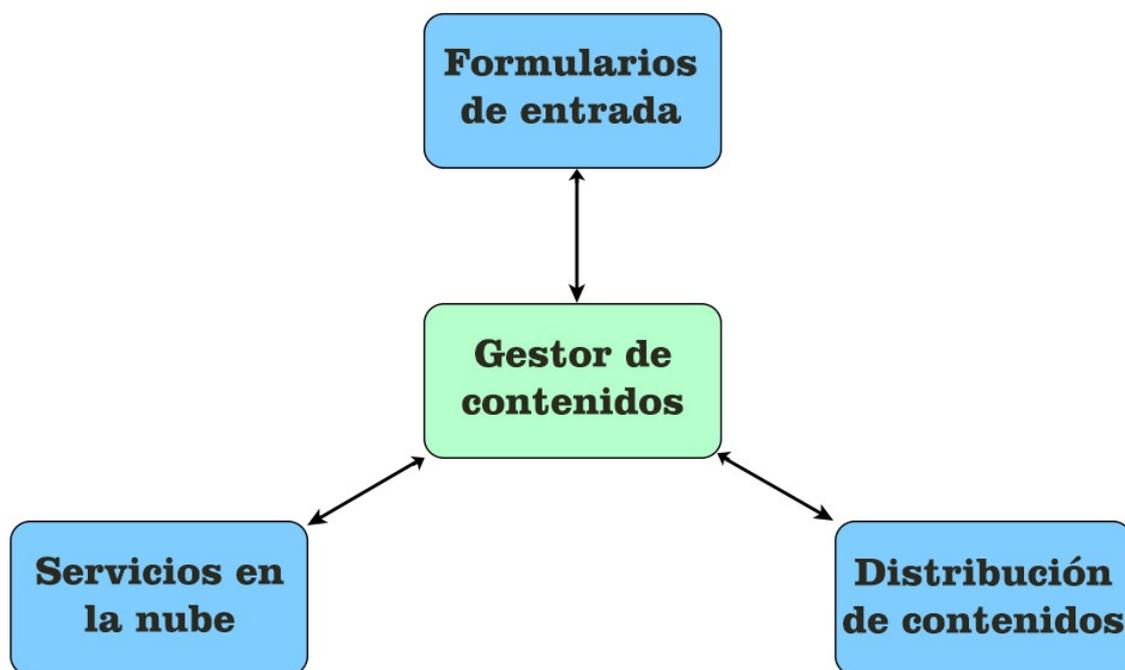


Figura 2.1 Arquitectura inicial

- **Bloque de servicios en la nube.** En este bloque se va a decidir qué tipo de alojamiento será el utilizado para la web y cual creará la interconexión entre el portal web y la aplicación móvil para que sirva de fuente de datos para la aplicación. El alojamiento del portal web se va a realizar utilizando los servicios que ofrece la nube, es decir, se va a hacer uso de diferentes servicios a través de Internet, desechando otras opciones más habituales

como son los servidores en alojamientos compartidos, dedicados o montando un servidor propio. Las razones para elegir los servicios en la nube se deben a dos razones fundamentales:

1. Utilizar un nuevo procedimiento de alojamiento web para así tener conocimiento sobre él y ampliar el conocimiento sobre alojamientos web habituales, y así trabajar bajo un nuevo paradigma que representa un cambio en el modo de creación e implementación de aplicaciones web.
2. Realizar un proyecto a coste cero, o de menor gasto posible, que abre nuevas opciones de trabajo para desarrolladores que están comenzando o no disponen de suficiente poder económico para afrontar grandes inversiones.

Los servicios en la nube o computación en la nube es una forma de trabajo relativamente nueva y en expansión y que ofrece una gran y variada oferta de mercado por lo que una vez decidido que se va a utilizar los servicios en la nube, se procede al estudio de los servicios en la nube que ofrece el mercado para elegir los más adecuado para la realización del proyecto.

- **Bloque de distribución de contenidos.** Este bloque es el dedicado a crear un servicio que ofrezca el almacenado y distribución del contenido estático. Principalmente este contenido estará compuesto por imágenes. Este servicio debe servir de fuente de datos tanto al portal web como a la aplicación móvil. Para ello se realizará un estudio para elegir el más adecuado para el proyecto basado en la relación calidad del servicio frente al precio por utilizar dicho servicio.

2.1 Gestor de contenidos

CMS son las siglas de Content Management System, que se traduce directamente al español como Sistema Gestor de Contenidos. Centrándonos en

este caso particular se utiliza un gestor de contenidos web, que se define como una herramienta que contiene un sistema de gestión para la creación y manejo de páginas web. Entre los más conocidos están Joomla, Drupal y Wordpress. Generalmente los WCM tienen una interfaz sencilla para permitir que un usuario pueda administrar y actualizar una web sin tener que ser un experto en los lenguajes de programación destinados a la creación de pequeñas páginas web.

Los sistemas WCM son un gran avance para la creación de los portales web para pequeñas empresas y organizaciones ya que proporciona una gran herramienta a bajo coste, pudiéndose centrar en los contenidos que se van a incluir en ellos.

2.1.1 Distintos CMS del mercado

En la actualidad, la oferta de gestores de contenidos web es bastante amplia. El mercado ofrece gran variedad con respecto al mundo de los gestores de contenidos. Como es una tarea extensa nombrar y comentar las características de cada uno de ellos el estudio se centrará en los tres más conocidos, que son WordPress, Joomla y Drupal.[1]

Según un estudio de mercado realizado el 22 de Septiembre de 2014 sobre la elección de los distintos gestores de contenidos web a la hora de crear un portal web[2], la imagen 2.2 expuesta en la siguiente página revela que los tres gestores de contenidos más usados son WordPress, Joomla y Drupal y tienen una cuota de mercado conjunta aproximadamente del 60 %, repartidas entre el 47'66 % de WordPress, el 6'26 % de utilización de Joomla y Drupal que copa un 5'75 % de la cuota de mercado de los gestores de contenidos web. En el gráfico mostrado a continuación se observa que Blogger copa una cuota de mercado de casi un 11'50 %, pero con la particularidad de que al no dar la opción de crear y modificar código de programación limita mucho la acción de los desarrolladores por lo que esta opción es descartada para la realización del portal web.

La filosofía de los gestores de contenidos es similar en todos los casos, pero existen algunas diferencias que hacen que los usuarios prefieran uno con respecto a otro.

- **WordPress.** Inicialmente creado para blogs su popularidad no ha dejado de crecer desde su creación. Uno de los mayores atractivos es su facilidad de uso y rápido aprendizaje. Cuenta con una amplia comunidad de usuarios y existe bastante documentación oficial.

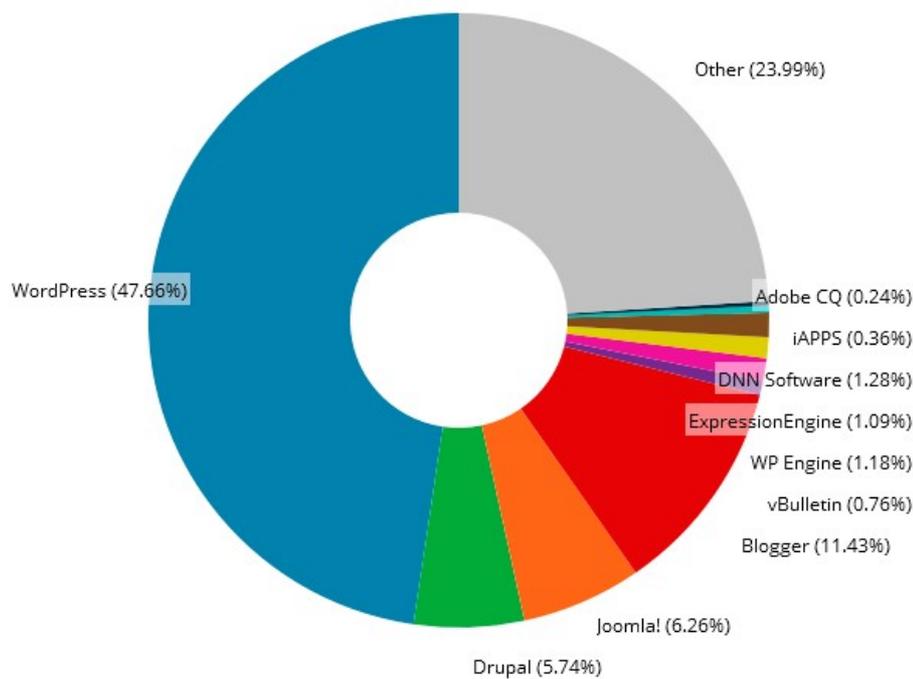


Figura 2.2 Cuota de mercado de WCM[2]

- **Drupal.** Se autodenomina una solución robusta y potente capaz de dar respuesta a cualquier necesidad funcional. Tiene un proceso de aprendizaje más complicado que las otras debido a su interfaz de administración, que no es fácil de usar a primera vista. Drupal tiene la posibilidad de crear varios roles diferentes de usuarios dotando a cada uno los privilegios que deseemos de forma personalizada. El diseño de Drupal es especialmente idóneo para construir y gestionar comunidades en Internet.
- **Joomla.** Tuvo su auge anterior a WordPress pero éste le ha quitado la mayoría de la cuota de mercado. Es de código abierto como WordPress y Drupal. Una de los mayores problemas que tiene es la seguridad, sobre

todo entre las extensiones que ayuda a añadir funcionalidad a la instalación básica.

Comparativa de WCM			
Característica	WordPress	Drupal	Joomla
Lenguaje	PHP	PHP	PHP
Precio	Gratis	Gratis	Gratis
Código abierto	Si	Si	Si
Cuota de mercado	47,66 %	5,74 %	6.26 %
Dificultad del programador	Baja	Media	Alta
Dificultad del administrador	Baja	Alta	Media
Curva de aprendizaje	Baja	Alta	Media
Flexibilidad	Alta	Muy alta	Baja
Documentación	Excelente	Alta	Baja
SEO (Search Engine Optimization)	Excelente	Malo	Medio

Tabla 2.1 Comparativa de los WCM

En la tabla 2.1 se muestran las características de cada uno de los gestores de contenido web. Los tres tienen en común que son de código abierto, gratuito y programados en el lenguaje de programación PHP. Las características de gestión y administración varían, y cada uno tiene sus ventajas e inconvenientes. En general Drupal es el que plantea más dificultades para el programador y el administrador a la hora de sacarle rendimiento óptimo, pero que compensa con una alta flexibilidad una vez se haya pasado la etapa de aprendizaje. Por el contrario WordPress es el más accesible para los nuevos desarrolladores por su menor dificultad y la calidad y gran cantidad de documentación tanto oficial como la aportada por sus usuarios a través de foros y webs. En tercer lugar tenemos a Joomla, que en principio es la opción menos adecuada ya que a una menor flexibilidad que los otros dos WCM para los desarrolladores se unen una dificultad media mayor tanto para programadores como administradores y una deficiente documentación.

La elección de un gestor de contenidos web por encima de otro viene dada, además de por las características antes nombradas por la funcionalidad del portal web que se va a crear. Si se va a crear una web corporativa no excesivamente compleja o de noticias, la elección debería ser para WordPress o Joomla, si por el contrario se va a crear un portal web con un uso empresarial más importante donde prime la escalabilidad y la potencia por encima de la estética y la facilidad de uso la opción recomendada es Drupal.

Una vez estudiadas las características principales de los WCM más importantes, se toma la decisión que para la creación del portal web del proyecto fin de carrera se elige la opción del gestor de contenidos web WordPress por las siguientes razones:

- Ser un WCM gratuito y de código abierto. Esta última propiedad permite modificar cualquier elemento según convenga al desarrollador.
- Ofrecer a los usuarios de WordPress una documentación oficial que contiene una guía de ayuda extensa y clara.
- Ser el gestor de contenidos más popular por la comunidad de desarrolladores, que implica tener gran cantidad de información adicional a la oficial.
- La teórica menor dificultad para programadores noveles que comienzan a desarrollar portales web y su amigable interfaz de administración. Todo ello tiene como resultado una menor curva de aprendizaje por lo que se sacará un rendimiento óptimo en un menor espacio de tiempo.
- Tener una extensa oferta de aplicaciones que añaden funcionalidad extra y mejoran la interfaz del portal web, lo que en WordPress se denomina *plugins* y *themes*.
- Ser el WCM que mejor se adapta al modelo de portal web creado para el proyecto fin de carrera debido a que va a tener elementos semejantes a los de un blog de noticias, y es para lo que WordPress fue concebido.

El cometido principal de WordPress dentro del proyecto es la de crear el contenido que será accesible tanto en el portal web como en la aplicación móvil,

ésto es, la creación de los eventos culturales que compondrán la agenda cultural de Gran Canaria.

Otras tareas que atañen a WordPress son la de interconectar a través de los *plugins* los servicios en la nube que conformarán la arquitectura del sistema y crear la interfaz del portal web.

2.2 Creación de formularios para el portal web

La otra sección dentro del capítulo de los desarrolladores que se encargan de crear y manejar el contenido dentro del portal web es la creación de los formularios para que los usuarios puedan interactuar con el portal web.

Para realizar estos formularios se utiliza un *framework* bajo el patrón MVC(modelo-vista-controlador). Un *framework* es un marco de trabajo que facilita a los desarrolladores crear software de manera más organizada dentro de otro proyecto de software, en otras palabras, un software que ayuda a la creación de software. El patrón MVC permite que cada elemento se trate por separado por lo que ayuda a su organización y mantenimiento.

Debido a estas propiedades de organización y de mantenimiento unidas a la reutilización de código hacen que la utilización de un *framework* bajo el patrón modelo-vista-controlador sea la opción adecuada para la creación de los formularios a utilizar dentro del portal web.

El *framework* elegido para la creación de los formularios es Zend Framework 2. La elección de Zend Framework 2 está justificada por varios motivos:

- Experiencia anterior del grupo de trabajo perteneciente a este proyecto fin de carrera.
- Ser el *framework* que más se aproxima al estándar PHP, ya que está desarrollado por los mismos creadores del código PHP.
- Ser un *framework* gratuito, que es una de las premisas iniciales para la creación de este proyecto.

Zend Framework 2 es la herramienta que se utiliza para crear el bloque anteriormente nombrado y llamado Formularios de entrada. Con Zend Framework 2 se va a crear un formulario para el registro de usuarios en el portal web y otro formulario para que los usuarios se pongan en contacto con la administración de la web ante la aparición de cualquier tipo de incidencia.

Se ha comentado anteriormente en la sección 2.1.1 que WordPress da la posibilidad de añadir funcionalidad extra a una web a través de los *plugins*, dentro de la variedad de *plugins* que ofrece WordPress se encuentra Wopzen2[3], que permite utilizar código creado con Zend Framework 2 dentro de los archivos de WordPress.

Haciendo uso de Wopzen2 se ha realizado el procedimiento para integrar los formularios creados con Zend Framework 2 dentro del portal web desarrollado con WordPress.

Como se muestra en la figura 2.3, la conexión entre WordPress y Zend Framework 2 para la realización de los formularios se hace a través del *plugin* Wopzen2.

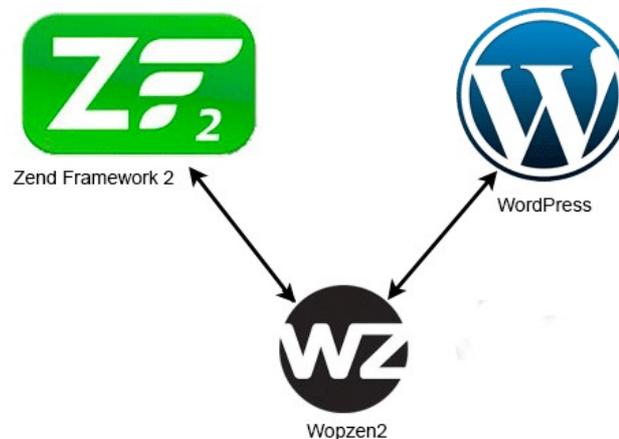


Figura 2.3 Conexión de WordPress y Zend Framework 2

Decididos los elementos que conforman las plataformas de desarrollo del portal web se procede a realizar el estudio de las aplicaciones en la nube que conforman el sistema para el alojamiento del portal web, la distribución y almacenamiento de contenidos y la conexión entre el portal web y el framework

encargado de la creación de la aplicación.

2.3 Computación en la nube

El NIST (*National Institute of Standards and Technology*) del gobierno de los Estados Unidos es uno de los principales usuarios de computación en la nube por lo que se toma como referencia a la hora de hablar de la computación en la nube (*cloud computing*). El *cloud computing* es un modelo tecnológico que permite el acceso ubicuo, adaptado y bajo demanda en red a un conjunto compartido de recursos de computación configurables compartidos (por ejemplo: redes, servidores, equipos de almacenamiento, aplicaciones y servicios), que pueden ser rápidamente aprovisionados y liberados con un esfuerzo de gestión reducido o interacción mínima con el proveedor del servicio.

El NIST a la hora de hablar de la computación en la nube asigna cinco características principales que éstos sistemas deben ofrecer:

- **Autoservicio bajo demanda.** El consumidor podrá aprovisionar recursos computacionales de forma flexible y unilateral, según lo requiera, y sin el requerimiento de interacción humana con el proveedor o proveedores del servicio.
- **Amplio acceso desde la red.** El acceso a los recursos de la nube está disponible desde cualquier lugar, en cualquier momento y desde cualquier dispositivo que pueda conectarse a una red IP. Simplemente se necesita una conexión a Internet.
- **Fondo de recursos.** Permite a los distintos proveedores compartir sus recursos entre los distintos usuarios, disminuyendo costes y maximizando la disponibilidad.
- **Escalabilidad y rapidez.** Gracias a las posibilidades de escalabilidad y elasticidad del *cloud computing*, es posible añadir o eliminar recursos en cuestión de minutos, aumentando o disminuyendo el almacenamiento o el número de procesadores sin que la aplicación se vea afectada.
- **Servicio medido.** Al cliente se le factura en función del nivel de servicios

proporcionados. Ésta es una de las características principales del cloud computing.

Además de estas características principales hay que considerar otra gran cantidad de ventajas que ofrece la computación en la nube: fiabilidad, calidad de servicio (QoS), costes más bajos o la facilidad de utilización.

Entre los inconvenientes que se pueden detallar del *cloud computing* están que el sistema no puede ser tan personalizable como se quisiera o que no se adapte totalmente a nuestras necesidades, la latencia inherente que conlleva, aunque las aplicaciones en la nube destacan en el procesamiento de datos a gran escala si nuestra aplicación necesita gran cantidad de transferencia de datos podría no ser el sistema más adecuado.

Otro inconveniente que plantea se refiere a la privacidad y seguridad que conlleva que nuestros datos al ser transportados y almacenados en un sistema ajeno debido a que se escapan a nuestro control y puede interceder la acción de *hackers* que puedan interceptar y robar nuestros datos.

La computación en la nube requiere de una infraestructura específica para su implementación y ésta se compone de dos capas. Primeramente está la capa física que se compone de todo el *hardware* necesario para soportar el servicio como servidores, componentes de red y discos de almacenamiento. La segunda capa es la capa de abstracción que consiste en el *software* implementado sobre el *hardware* que compone la capa física.

La computación en la nube se puede clasificar en dos grandes conjuntos de modelos distintos:

- **Modelo de implementación.** Se refiere a la ubicación y administración de la infraestructura de la nube, que puede ser pública, privada, híbrida o comunitaria.
- **Modelo de servicio.** Define el tipo concreto de servicios a los que se puede acceder en una plataforma de computación en la nube. Se ha estandarizado tres tipos de servicio, infraestructura como servicio (IaaS),

plataforma como servicio (PaaS) y software como servicio (SaaS).

La nube supone nuevas oportunidades para usuarios y desarrolladores, porque se basa en la posibilidad de acceder a recursos compartidos ofreciendo una serie de características de sistema que cambian totalmente la economía de la infraestructura de la tecnología informática y permite nuevos modelos empresariales.

Modelos de implementación

Define el propósito de la nube y la naturaleza del modo en que se ubica. La definición de NIST para los cuatro modelos es la siguiente:

- **Nube pública.** La infraestructura está disponible para el uso público de múltiples clientes y es propiedad de una organización que vende servicios a la nube. Proporciona ahorros en costes y gran flexibilidad para hacer frente a los picos de demanda por Internet.
- **Nube privada.** Exclusivo para el uso de una determinada organización. La gestión recae en la propia organización o en un tercero. Proporcionan mayor supervisión, seguridad y privacidad de los datos. Requiere de una mayor inversión.
- **Nube comunitaria.** Ofrecen una infraestructura compartida por varias organizaciones, normalmente de propósito común.
- **Nube híbrida.** Es una combinación de dos o más anteriores. El modelo de explotación genérico en este tipo de nubes es en el que las organizaciones utilizan la parte pública de la nube híbrida para servicios genéricos, reservando la parte privada para sus datos analíticos.

Modelos de servicio

- **Infraestructura como servicio, IaaS.** Este modelo proporciona al cliente aquellos servicios vinculados con las infraestructuras como los servidores o

almacenamiento de forma virtualizada. El cliente es responsable de todos los demás aspectos de la implementación, que pueden incluir el sistema operativo, aplicaciones e interacciones del usuario con el sistema, e interactúa con el modelo IaaS para completar su solución.

Otro aspecto fundamental a tener en cuenta, es que las infraestructuras como servicio pueden permitir una escalabilidad automática o semiautomática, de forma que podamos contratar más recursos según sea necesario. El ejemplo perfecto es el proporcionado por Amazon Web Service (AWS) que nos provee una serie de servicios como EC2 que nos permite manejar máquinas virtuales en la nube o S3 que es utilizado como almacenamiento.

- **Plataforma como servicio, PaaS.** PaaS proporciona la infraestructura de la nube, los sistemas operativos y el software necesario para que un cliente pueda desplegar sus aplicaciones en la infraestructura de la nube. El cliente es el responsable de la instalación y administración de la aplicación que se va a desplegar. Los sistemas PaaS deben ofrecer una forma de crear interfaces de usuario y ser compatibles con estándares como HTML o Javascript. Ejemplos populares son Google App Engine que permite desarrollar aplicaciones en Java desplegándolas en la infraestructura que provee Google, cosa que también sucede en la plataforma Heroku con el *framework* Ruby on Rails.
- **Software como servicio, SaaS.** Es un entorno operativo completo con aplicaciones, administración e interfaz de usuario. La aplicación se proporciona al cliente a través del navegador y la responsabilidad del cliente consiste en la administración de sus datos. Todo lo que va desde la aplicación hasta la infraestructura es responsabilidad del proveedor. El SaaS sería el equivalente en la nube al software que adquirimos al por menor por los medios habituales. Ejemplos populares de SaaS son Google Docs, Dropbox o Gmail.

Los tres modelos de servicio en conjunto se conocen como modelo SPI (Software, Plataforma e Infraestructura) de computación en la nube. Aunque

existen otros modelos como SaaS (almacenamiento como servicio), IaaS (identidad como servicio), etc; son los servicios SPI engloban todas las posibilidades.

Otro de los modelos que han ido apareciendo y que nace con las necesidad de dar soporte y vincular las aplicaciones web y móviles es el *Backend as a service* (BaaS). Es un modelo para proporcionar a los desarrolladores web y de aplicaciones móviles una forma de vincular estas aplicaciones al almacenamiento en la nube, servicios analíticos y/o otras características tales como la gestión de usuarios, la posibilidad de enviar notificaciones *push* o la integración con servicios de redes sociales.

Las aplicaciones móviles y web requieren unas características similares en su *backend* y el desarrollo de éstas por parte de los programadores puede ser una tarea costosa y extensa en el tiempo. Para liberar de ese trabajo a los desarrolladores han aparecido los proveedores de *Backend as a service* que ofrecen un servicio de puente entre la interfaz de usuario y la administración de la aplicación altamente escalable y sin inversión inicial importante.

Los servicios en la nube de *backend* son relativamente jóvenes ya que vieron la luz hace unos años y ha sido una revolución en el mundo de las aplicaciones debido a que ha facilitado en gran medida la tarea a los programadores y abierto la posibilidad de crear aplicaciones a los menos expertos. Se espera que haya un gran auge de estos servicios cuando los programadores se familiaricen con ellos y se den cuenta del gran potencial que tienen.

Han ido apareciendo muchas plataformas que ofrecen este servicio, entre las que se encuentran Parse, CloudMine, Sencha io, etc.

En la figura 2.4 se presenta a modo de resumen las posibilidades que ofrece la computación en la nube y los servicios que ofrece a los desarrolladores como servidores, base de datos, backend para aplicaciones, etc.

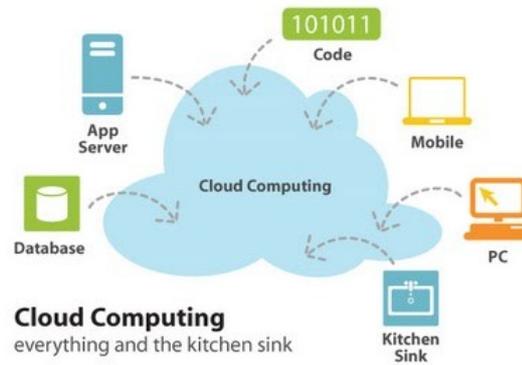


Figura 2.4 Cloud Computing

Cloud Computing en las empresas españolas

El uso de los servicios en la nube todavía no está muy extendido dentro de las empresas españolas.

Según un estudio obtenido por el Instituto Nacional de Estadística en Septiembre de 2014[4], representado en la figura 2.5, un 70 % de las empresas que tienen 10 o más empleados tienen un portal web. De estas empresas solamente un 15 % utilizaban soluciones de Cloud Computing a comienzos de 2014.

Dentro de este 15 %, la utilización que dan a los servicios en la nube son en un 69 % para almacenamiento de archivos, un 61,4 % para servicio de e-mail y un 54,7 % como bases de datos de la empresa. La nube pública como modelo de implantación fue la opción más elegida en un 53,4 % de las veces.

Las empresas que no hicieron uso de los servicios en la nube lo hicieron principalmente por dos motivos: el 46,5 % al no tener suficiente conocimiento de lo que son los servicios en la nube y el 45,9 % porque no creen necesario estos servicios para el desarrollo de la empresa.

En Canarias las empresas que utilizan el cloud computing es similar a la media española (15,1 %).

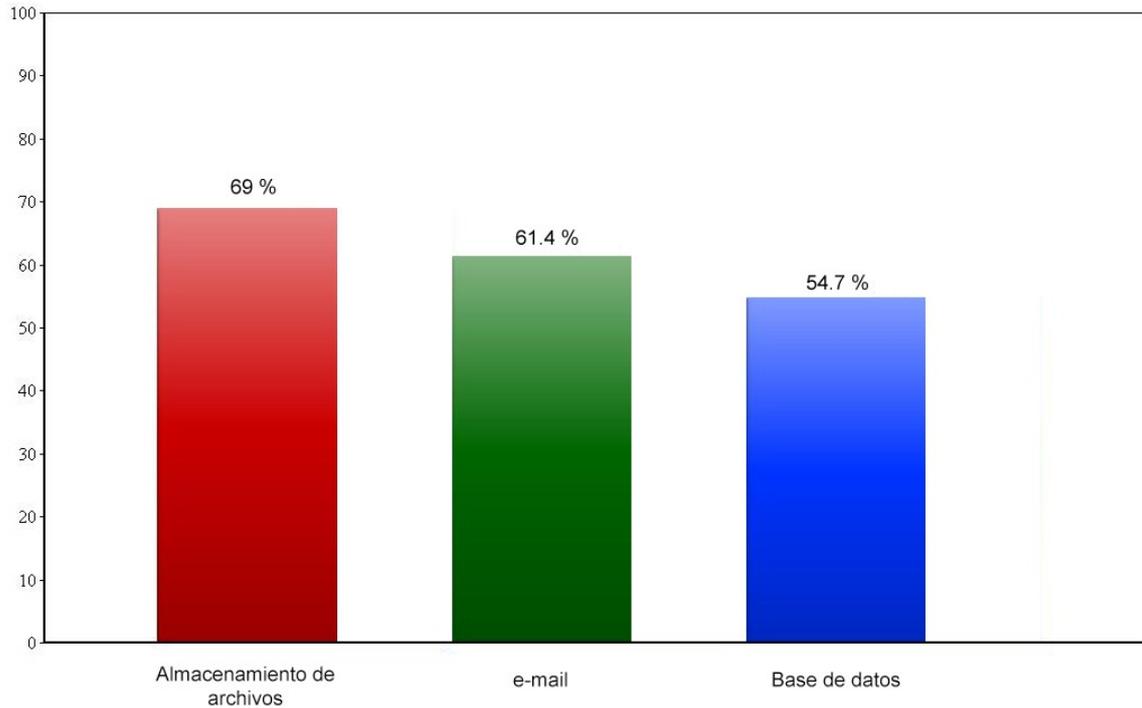


Figura 2.5 Uso del cloud computing por las empresas españolas

Con estos datos se saca la conclusión de que el cloud computing es un modelo de servicio totalmente desconocido por gran parte del empresariado español, por lo que hay que hacer un trabajo de información de los beneficios que acarrea este modelo, sobre todo a las pequeñas empresas que ayuda a su utilización debido a su bajo coste inicial.

2.3.1 Servicios en la nube utilizados

Alojamiento de la aplicación web

La primera decisión a tomar es la elección del tipo de servicio en el que alojar el portal web. Se trata de buscar el modelo de servicio adecuado donde realizar la instalación de WordPress.

La opción de utilizar una infraestructura como servicio queda descartada por la razón principal de que supone la forma más semejante a la contratación de un servidor para el alojamiento del portal web. En lugar de gestionar un servidor se alquila una infraestructura en la nube en la que se pone a disposición recursos

virtualizados donde instalar y personalizar la aplicación como se desee.

La elección escogida es la de utilizar una plataforma como servicio y más concretamente la plataforma Heroku en donde se realizará la instalación. Diversas razones han llevado a esta elección:

- Es la forma de trabajo en la que más se va a comprobar el procedimiento de trabajar con aplicaciones en la nube. Se verá más adelante en esta sección que será necesario la utilización de otros servicios en la nube para completar todos los servicios que ofrece el portal web.
- Considerada la mejor opción para desarrolladores que se inician en el campo de los servicios en la nube ya que se delega en la plataforma las tareas de administración del sistema operativo, servidores o máquinas virtuales.
- Elección ideal para pequeñas empresas o autónomos ya que su precio inicial es muy bajo, o gratuito en muchos casos, evitando así una gran inversión inicial. Muchas de ellas ofrecen limitación en espacio o tiempo, en el caso de la plataforma Heroku ofrece un plan gratuito para la instalación de aplicaciones que se adapta a las necesidades del proyecto limitando la capacidad pero no el tiempo de ejecución de la aplicación. Ofreciendo en el caso de necesidad futura escalabilidad de forma progresiva en tamaño y precio.
- Trabajar con un PaaS facilita la integración con servicios de terceros que es una operación habitual ya que ayuda a incorporar funcionalidades extras a las aplicaciones.
- Heroku permite realizar aplicaciones bajo el lenguaje de programación PHP que es el lenguaje utilizado para programar en WordPress.

Una particularidad en términos de computación en la nube que tiene el utilizar la plataforma Heroku como PaaS es que a su vez esta plataforma utiliza la Infraestructura como servicio EC2 de *Amazon Web Service* para tener gestionar sus recursos por lo que indirectamente también se está empleando un IaaS.

Almacenamiento de contenido

Una de las particularidades que tiene utilizar Heroku como plataforma para la instalación de WordPress es que el contenido generado por la web debe ser almacenado externamente.

El contenido generado, y que debe ser guardado por el portal web, es principalmente la inclusión de imágenes que aporten información extra a la creación de eventos.

El procedimiento empleado para el guardado de imágenes consiste en utilizar un servicio de almacenamiento de archivos combinado con una red de entrega de contenidos (CDN) que sirva para guardar y operar con los archivos necesarios para el sistema. Para realizar esta tarea se han utilizado los servicios ofertados por *Amazon Web Service*.

La empresa Amazon dentro del catálogo de servicios que ofrece en la nube está el servicio de almacenamiento Amazon S3 (Simple Storage Service) y el servicio CDN llamado Amazon CloudFront. Estos dos servicios combinados serán los encargados de guardar el contenido generado y entregarlo en el momento que sea demandado.

Para utilizar estos servicios en el sistema hay que realizar las dos siguientes operaciones:

- Crear una cuenta en Amazon y configurar los servicios de S3 y CloudFront. La creación y configuración de ambos servicios estará detallada en el capítulo 4.
- Utilizar un *plugin* de WordPress que permita conectar WordPress con el servicio de CloudFront de Amazon para guardar el contenido y obtenerlo cuando sea necesario. El *plugin* utilizado para ello se llama Amazon Web Services for WordPress.

Conexión con un BaaS

La arquitectura de este PFC contiene un servicio en la nube más. Es el

Backend as a service (BaaS), este modelo sirve para vincular una aplicación web y una aplicación móvil que comparten información. Las razones para utilizar un BaaS son:

- Es la plataforma en la nube creada para interconectar las aplicaciones web con las aplicaciones móvil.
- Su precio. Las plataformas que ofrecen BaaS ofrecen servicios gratuitos para comenzar a trabajar con este método de trabajo.
- Escalabilidad. Como todos los servicios en la nube facilitan la escalabilidad en el caso de que sean necesarios más recursos debido al aumento de la demanda.
- Seguridad. Al tratarse de grandes empresas orientadas al trabajo en la nube tienen un gran sistema de seguridad ante ataques externos, lo que repercute en tener un mejor sistema de seguridad en las aplicaciones.
- Multiplataforma. Es independiente de la plataforma móvil con la que se va a trabajar.

La utilización que se va a hacer del BaaS en el proyecto tiene como finalidad guardar los datos generados en el portal web y almacenarlos en su base de datos para que éstos sean accesibles desde la aplicación móvil. Existen dos tipos de datos que se van a guardar en BaaS:

- Los datos de los usuarios que se den de alta en el portal web.
- Los datos de los eventos creados en WordPress para que sean accesibles por el framework y poder mostrarlos en la aplicación móvil.

Para realizar estas operaciones se va a utilizar Parse, que es uno de los BaaS que ofrece el mercado. Se crean varias tablas donde se almacenarán los datos generados correspondientes a los datos de usuarios y a los eventos de la agenda cultural. Estas tablas estarán más detalladas en el apartado dedicado a Parse dentro del capítulo 4 donde se amplía la información correspondiente a los servicios en la nube utilizados en el proyecto fin de carrera.

Parse como BaaS oferta un plan gratuito mensual a los desarrolladores de aplicaciones que cumple con las necesidades del proyecto:

- Un millón de peticiones al mes.
- Un máximo de treinta peticiones por segundo.
- 1 GB de espacio para almacenar archivos.
- Posibilidad de crear notificaciones push, comunicación desde el servidor al cliente sin necesidad de acción por parte del cliente.

Ejecutar la conexión entre WordPress y Parse para el envío bidireccional de los datos se va a producir mediante una conexión REST (Representational State Transfer).

Una conexión REST es una conexión que se accede a través del protocolo HTTP, es decir, se accede a través de la especificación de una URL en la que se envían los datos. Con este método se interactúa con Parse para la inserción, modificación, petición o eliminación de datos de entre sus tablas.

Para realizar una comunicación REST con Parse hay que seguir unas pautas:

- El acceso debe ser a través del dominio de la API de Parse. Una API (Application Programming Interface) es un conjunto de funciones que permiten acceder a servicios proporcionados por un tercero.
- En el caso de inserción de datos en Parse, codificación a través del código JSON de los datos a insertar a través del par nombre-valor.
- Indicar la operación a realizar mediante los métodos *GET*, *POST*, *PUT* o *DELETE* para crear, insertar, obtener o borrar datos en Parse.

Conexión con las aplicaciones móviles

La utilización de Parse como BaaS del sistema tiene como razón principal la existencia de un framework creado por el Instituto Universitario de Microelectrónica Aplicada (IUMA) que realiza la tarea de obtener e interpretar los

datos almacenados en Parse y crear aplicaciones móviles en los sistemas operativos Android e IOS.

Para realizar la conexión hay que crear unas tablas y campos específicos donde guardar los datos de un determinado formato para que sea posible la interconexión Parse-framework para la creación de las aplicaciones móviles.

En la figura 2.6 se muestra a modo de resumen la arquitectura creada para el Proyecto Fin de Carrera:

- WordPress es el gestor de contenidos elegido para crear el contenido del sistema.
- WordPress se instala en una Plataforma como servicio como es Heroku.
- Se crean formularios con Zend Framework 2 y se integran dentro de WordPress a través de un *plugin*.
- Para almacenar y enviar el contenido a los usuarios se utiliza una red de CDN con los servicios de Amazon Web Service.

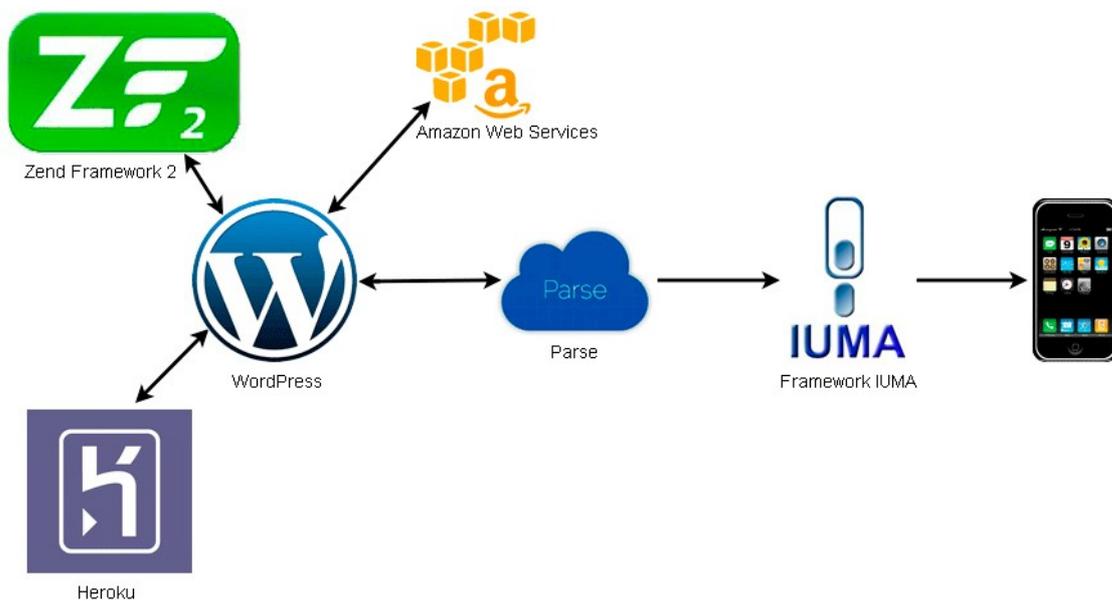


Figura 2.6 Resumen de la arquitectura del pfc

- Se hace uso de un Backend as a service como Parse que sirve para de nexo de unión entre el portal web y la aplicación móvil. Para realizar esta

conexión hacemos uso de código en JSON.

- Una vez los datos están en las tablas de Parse se acceden a ellos a través del framework creado por el IUMA que interpreta los datos ahí guardados para crear la aplicación móvil.

Plataformas de desarrollo

Una vez hecho el estudio de las plataformas más adecuadas para la creación de contenidos se procede a explicar con más detalle cuales son las plataformas utilizadas.

La primera parte se corresponde a desarrollar el funcionamiento y comentar las diferentes partes del gestor de contenidos WordPress. La administración de WordPress ocupa la primera parte del capítulo, mostrando y explicando las partes fundamentales para navegar dentro del gestor de contenidos. La creación de contenido dentro de WordPress finaliza la demostración del funcionamiento del gestor de contenidos.

La segunda parte está dedicada a Zend Framework 2, el marco de trabajo elegido para la creación de los formularios del portal web. Se detallará el patrón modelo-vista-controlador, los frameworks que trabajan bajo este patrón y finalmente se comentará Zend Framework 2 y cómo es utilizado dentro del proyecto.

3.1 WordPress

WordPress es el sistema gestor de contenidos más popular, de hecho actualmente existen unos 12 millones de portales web que trabajan alrededor del mundo con WordPress[1]. Fue ideado en 2003 por Matt Mullenweg para crear un servicio orientado a creación de blogs de forma elegante y bien diseñados. Debido a que es un proyecto *Open Source*, de código abierto, cientos de personas colaboraron a su desarrollo y expansión por lo que su fama se disparó y fueron muchos los que migraron de otras plataformas para crear sus trabajos en WordPress. Una de las partes más significativas de WordPress es que tiene muy bien diferenciado la parte del diseño con la parte del contenido.

A raíz de su gran expansión, WordPress se ha dividido en dos ramas muy bien diferenciadas. Una es Wordpress.com y otra es Wordpress.org, que son dos conceptos totalmente distintos.

Wordpress.com

En wordpress.com un usuario crea su blog *online* sin tener que preocuparse del sistema de gestión de la web, almacenamiento de ficheros, actualizaciones, etc. Es la forma más sencilla que tiene WordPress de ofrecer su servicio al usuario ya que éste no necesita tener prácticamente conocimiento alguno sobre programación.

WordPress ofrece al usuario diversos estilos para el diseño del blog para que decida según su criterio cuál se adapta mejor a sus necesidades y gusto. En lo único que se tiene que centrar es en la edición de contenido, al que accede a través de un panel de administración.

En este panel destaca por encima de todos la creación de *posts*, que es como se conoce a la creación de una entrada en el blog. A través de un editor de texto similar a los que se utilizan habitualmente en los programas de edición de textos se procede a la creación de las entradas, con la posibilidad de incluir fotos, enlaces y toda una serie de complementos que ayudan a crear un entrada de blog elegante y funcional. Otra de las características fundamentales es la creación de categorías y etiquetas que ayudan a identificar y localizar tu blog tanto a otros usuarios como a los buscadores.

En definitiva, wordpress.com ofrece un servicio de blogging a usuarios poco avanzados en el manejo web, ya que ellos se encargan de la gestión de los recursos dejando al usuario que se encargue del contenido a tratar.

Wordpress.org

La principal característica y que la diferencia de wordpress.com es que el usuario se encarga de la instalación, mantenimiento y actualización del sistema. Aquí son necesarios unos conocimientos más avanzados para poder crear un portal web. Ahora el usuario no se encarga únicamente del contenido, además tiene que proporcionar un espacio en la red donde alojar su web.

Debido a que ahora se tiene control sobre todas las partes que se manejan en una web tenemos la capacidad, al ser de código abierto, de modificar cualquier

parte del código para adaptarlo a nuestras necesidades, ésta es la gran ventaja para poder crear portales web a la medida. Se tienen infinitas funcionalidades y posibilidades, como poder tener mayor capacidad de almacenamiento o la posibilidad de incluir publicidad. Un desarrollador puede hacerle tantos cambios como desee, tanto de apariencia como de contenido.

Para trabajar sobre la apariencia de la web se cuenta con dos herramientas principales, los *themes* o plantillas y los *plugins*.

Las plantillas de WordPress definen la estructura y apariencia de una web, y son independientes del contenido que queramos incluir en ellas. Hay miles de plantillas en WordPress ya que hay una gran comunidad de usuarios que se dedican a crear plantillas tanto gratuitas como de pago. Existen plantillas de todo tipo desde las más básicas para diseñar un blog hasta la web más completa que nos podamos imaginar.

Los *plugins* son pequeñas aplicaciones que incluimos para añadir funcionalidad a nuestra web. Estas pequeñas aplicaciones pueden hacer las más diversas funciones, como puede ser la creación de una galería fotográfica, la creación de un formulario o ayuda para el posicionamiento web.

La incursión de contenido es similar al anterior. WordPress provee de un editor de texto para crear contenidos que se van a alojar en la red, aunque existe la posibilidad de enriquecer nuestra información ya sea a través de los mencionados *plugins* o a través de la modificación o inclusión de código propio.

En la parte de mantenimiento de la web el usuario tiene que buscar un dominio y un *hosting* donde alojar la web. Además, el usuario se encargará de la seguridad de la web y de la actualización de plantillas o *plugins* por lo que se corre el riesgo de no hacerlo correctamente y la web permanezca sin funcionamiento. Es el contrapunto a tener mayor libertad y poder modificar toda la web al gusto del usuario.

La creación del portal web para el proyecto fin de carrera se realiza haciendo uso de la instalación del software a través de wordpress.org por los siguientes motivos:

- No va a ser únicamente un blog donde se creen entradas.

- Se va a modificar e incluir código para añadir funcionalidad al portal web como la de conectar con Parse.
- Se incluyen *plugins* para crear un portal web más completo.

Para el aprendizaje del funcionamiento de WordPress se realizó una instalación de forma local en un ordenador personal antes de trabajar con la plataforma Heroku para así tener un entorno seguro donde realizar todo tipo de pruebas sin temor a perder información valiosa y hacerlo de una forma rápida. Estas pruebas consistieron en:

- Estudio de la creación y edición de Entradas.
- Estudio de la creación y edición de Páginas.
- Estudio relacionados con los temas de WordPress. Qué es un tema y elección del tema adecuado a nuestras necesidades.
- Estudio de los *plugins* de WordPress. Qué son los *plugins* y estudio de los *plugins* necesarios para el funcionamiento del portal web.
- Crear código PHP que añada funcionalidad extra al portal web y mejore la interfaz de usuario.

3.1.1 Administración de WordPress

Finalizada la instalación y puesta en marcha de WordPress, dentro del Pliego de condiciones de este documento, se procede a describir el panel de administración de WordPress. En este panel de administración se encuentran todas las opciones que ofrece WordPress para configurar el portal web que se quiere realizar. Navegando por su menú se hallan las opciones de creación y edición de contenidos, añadir funcionalidades adicionales al sitio web o editar código para moldear la interfaz y adaptarlo a los requisitos de la web.

Este panel de administración difiere dependiendo si el que accede a la administración de WordPress es un administrador del sistema o un editor de contenidos. Al administrador se le otorgan todos los permisos posibles, por lo que

puede modificar cualquier elemento, mientras que los editores sólo tendrán la opción de crear y editar el contenido.

Los próximos apartados se adentrarán en mostrar el menú de administración de WordPress para el administrador ya que es el que más funciones tiene y que incluyen las funciones que desarrollan también los editores de contenidos.

- **Escritorio**

El escritorio es el primer sitio al que se accede una vez se está identificado en WordPress. La imagen 3.1 muestra cómo es el escritorio de WordPress cuando se accede al sistema. Aquí WordPress ofrece información sobre la actividad reciente de la web, la plantilla se que está utilizando, las entradas publicadas, si hay alguna actualización pendiente, etc. Además ofrece un menú vertical a la izquierda con las diferentes opciones que se pueden manejar en nuestra instalación de WordPress. Las más importantes de ellas se explicarán a continuación.

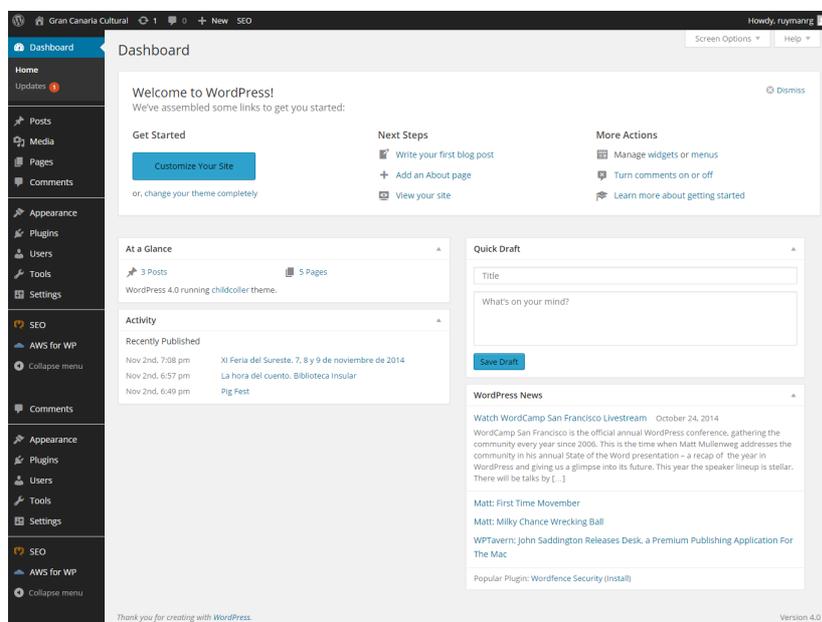


Figura 3.1 Escritorio de WordPress

- **Entradas**

Se define así a cada uno de los artículos que se quieren compartir en el sitio web y que se mostrarán en orden cronológico inverso en la página de inicio de la web. También son llamados como su término inglés *posts*. Es el método para crear contenido dentro de WordPress. Estas entradas pueden contener además de texto imágenes, audio o vídeo. Cada entrada tiene:

- **Título.** El título de la entrada que se va a publicar.
- **Texto.** El cuerpo de la entrada a publicar.
- **Categoría.** Cada entrada se clasifica en una o varias categorías.
- **Etiquetas.** Palabras que definen el contenido de una entrada.
- **Enlaces.** A una imagen o cualquier otro objeto.

Una vez se accede al menú Entradas allí están visibles todas la entradas o *posts* creados anteriormente para ser modificados o borrados, crear una nueva entrada para la web o la creación de categorías y etiquetas para clasificar las entradas creadas.

En el portal web del proyecto, la creación de los eventos que constituyen en conjunto la agenda cultural son cada una de las entradas creadas en WordPress.

Para la clasificación de las entradas creadas se utilizan las categorías, de este modo los eventos están organizados por los veintidós municipios de Gran Canaria, para que sea más fácil su búsqueda por parte de los usuarios de la web y de la aplicación móvil.

Esta sección del menú es la única de relevancia que tienen en común un administrador y un editor ya que es la que concierne a la gestión de contenidos dentro del sistema. Pero debido a los permisos que éstos tienen hay diferencias entre las funciones a realizar por uno y por otro.

Un administrador tiene los permisos para acceder, modificar y borrar cualquier entrada, ya sea propia o de cualquiera de los editores registrados en el sistema. Por el contrario, un editor tiene la posibilidad de ver todo el contenido

creado en el sistema pero sólo podrá editar y borrar aquel contenido creado por él mismo.

Por lo general esta sección será la utilizada por los editores para gestionar el contenido del sistema, mientras que el administrador, aunque tenga acceso a gestionarlo su cometido no está tanto en el contenido sino en gestionar otras partes del sistema.

- **Páginas**

Una página muestra información a los usuarios y se diferencia principalmente de una entrada por ser más estática en cuanto a contenido. La información que suele contener, por norma general no es actualizada con frecuencia, por lo que es ideal para introducir información que no pierde importancia con el tiempo como por ejemplo información sobre el autor de la web, página de contacto, índice de contenidos, etc.

- **Apariencia**

En el apartado de apariencia tenemos varias categorías que se van a configurar para crear una interfaz de usuario amigable a las personas que van a ser uso del portal web.

Temas

Éste es junto con las entradas uno de los apartados más importantes de WordPress, ya no sólo de la apariencia sino que podemos decir que de todo el universo WordPress, ya que aquí se va a elegir la apariencia que tendrá el portal web.

La plantilla o *theme* de WordPress elegida para la creación de la página web se llama *Coller*. Esta plantilla ha sido elegida en parte por su estética ya que se adaptaba a lo planteado en el boceto inicial, pero principalmente por su adaptabilidad a las necesidades de la web:

- Diseño web adaptable (*Responsive web design*). Calidad que hace que la web se adapte al entorno del usuario (ordenador, tablet, móvil, etc.)
- Plantilla gratuita. Premisa inicial del proyecto.

Widgets

Los *widgets* añaden funcionalidad a los portales web mostrando información principalmente en las barras laterales de la web. WordPress ofrece por defecto varios de estos *widgets* como puede ser un buscador, las entradas publicadas más recientes o un reloj.

Es habitual que los *plugins* incluyan también *widgets* y eso es lo realizado en la web añadiendo un *widget* a la barra lateral derecha del *plugin* Google Calendar Events.

Este *plugin*, como se explicará con más detalle posteriormente, añade la posibilidad a los usuarios de sincronizar los eventos con el Google Calendar. La función que realiza el *widget* es que los eventos más próximos a su celebración sean mostrados en todas las páginas del portal web para que sean más visibles a los usuarios.

Menú

WordPress ofrece la posibilidad de añadir cuantos menús se quieran a un portal web, un menú principal encabezando la web, menús verticales a ambos lados y hasta un menú a pie de página. Los menús pueden estar formados por páginas, enlaces o categorías según más convenga.

Un único menú principal es lo que contiene el portal web creado. En este menú se incluyen las distintas secciones que conforman la web.

- **Plugins**

Los *plugins* o complementos son aplicaciones que se instalan en

WordPress para añadir funcionalidad a una web para hacer que tenga más utilidad que la de ser un blog, y es lo que ha hecho que WordPress se haya convertido en algo más que un CMS para crear blogs.

Existen cientos de miles de *plugins* con los que se puede implementar casi cualquier necesidad que un desarrollador tenga. Aunque los *plugins* son beneficiosos ayudando a crear un portal más complejo se recomienda no abusar de ellos debido a que pueden influir negativamente en la velocidad y carga de una web .

En el trabajo realizado se han utilizado algunos *plugins* que se han creído fundamentales para la realización del mismo y que la alternativa de crear estas funcionalidades a través de código propio en PHP suponían un gran esfuerzo de trabajo y tiempo.

A continuación, se nombran y explican los *plugins* utilizados y la labor que desempeñan.

- **Wopzen2.** Este *plugin* permite incluir el código creado con Zend Framework 2 dentro de las páginas PHP de WordPress. El *plugin* crea la estructura con la que trabaja Zend Framework 2 dentro del sistema de WordPress. A través de Wopzen2 se incluyen en las páginas registro.php y contacto.php creadas en WordPress los formularios para el registro de usuarios y el contacto de los usuarios con la administración de la aplicación.
- **Google Calendar Events.** Permite sincronizar el calendario incluido en la web con uno o varios calendarios de Google (Google Calendar). A través de este *plugin* los administradores importan los eventos creados en Google Calendar para que sean visibles a modo de calendario en la web. Además los visitantes de la web pueden exportar dichos eventos e incluirlos en su propio Google Calendar y hacer de ellos el uso que quieran.
- **Amazon Web Services for WordPress.** Este *plugin* nos permite interconectar WordPress con el servicio Cloudfront de Amazon. Este *plugin*

guarda los archivos de un sitio web en los servidores remotos de Amazon, y sirve estos archivos a los usuarios desde múltiples lugares en el mundo para optimizar el tiempo de carga y entrega de archivos.

La utilización de este *plugin* se realiza para el almacenamiento de las imágenes que son necesarias en la aplicación. Principalmente para el almacenamiento de todas las imágenes que serán incluidas en las entradas de los eventos.

- **WordPress SEO by Yoast.** Este último *plugin* nos ayuda a configurar la web para mejorar en el posicionamiento web o SEO. Configurar el posicionamiento de una web es primordial para aparecer en las primeras posiciones de los buscadores y así ser más visible para los usuarios de internet.
- **Usuarios**

La sección de usuarios en el menú de WordPress es utilizada por el administrador del sistema para administradores todos los usuarios que interaccionan con el sistema.

Al crear una nueva instalación de WordPress se crea un usuario que es el administrador creado por defecto en el momento de la instalación. Este usuario puede crear nuevos usuarios dándole diferentes roles como se explicó en la parte de la memoria que abordaba el tema de los CMS. Cabe la posibilidad de crear un nuevo usuario que sea administrador del sistema, editor de contenidos, colaborador o subscriptor.

La recomendación es que haya uno o dos administradores que puedan manejar el sistema completo ante la necesidad de realizar cambios y un número mayor de editores que se encarguen de crear el contenido de la aplicación.

3.1.2 Child Themes

Un *child theme* o "tema hijo" es un tema que hereda WordPress la

funcionalidad de otro tema, llamado "tema padre". Los temas hijos permiten modificar o añadir las funcionalidades del tema padre. Un tema hijo es la forma más segura y fácil de modificar un tema existente, tanto si se quiere hacer pequeños cambios como otros más amplios. En lugar de modificar los archivos del tema directamente, es posible crear un tema hijo y hacerlo prevalecer sobre el tema padre.

La conveniencia de crear un *child theme* es un tema de debate entre los programadores que utilizan WordPress debido a que existen corrientes de opinión a favor y en contra. Entre las ventajas de crear un tema hijo están:

- Si modificas un tema existente y se actualiza, tus cambios se perderán. Con un tema hijo, puedes actualizar el tema padre (que puede ser importante por cuestiones de seguridad o funcionalidad) y seguir manteniendo los cambios.
- Puede acelerar el tiempo de desarrollo.
- Es una buena forma de comienzo si se está empezando a aprender sobre desarrollo de temas WordPress.

Las opiniones que están en contra de crear temas hijo destacan las siguientes desventajas:

- Como un *child theme* realiza una llamada a otro tema, el superior, que es de donde obtiene la mayoría de su código, esto puede afectar al rendimiento de tu web, ralentizándola y afectando al posicionamiento web.
- Se requiere una visión global y exhaustiva de todo el código padre para no empeorar el aspecto de una web.

Para la realización de este proyecto se decidió la creación de un tema hijo principalmente por dos motivos:

- El primero y principal como recomienda la página oficial de WordPress no es otro que el aprendizaje sobre el desarrollo de temas de WordPress, así

se está en disposición de poder hacer todos los cambios que se consideren oportunos en el tema elegido de forma segura y de fácil arreglo.

- El segundo motivo por el que se decantó a utilizar un tema hijo es para guardar los cambios realizados ante una posible actualización del tema principal.
- **Creación de un tema hijo**

Se crea una carpeta en la carpeta de temas de WordPress para albergar el tema hijo. La carpeta de temas está en *wp-content/themes*. Se debe nombrar la carpeta sin ningún espacio como parte del nombre, y es habitual usar el nombre del tema padre con el prefijo "hijo-" (o "child-" en inglés) adjunto a él. Por ejemplo, en este caso al estar utilizando el tema *coller*, la carpeta será nombrada como *childcoller*.

En la carpeta del tema hijo, se crea un archivo que se llame *style.css* que es la hoja de estilo que se utiliza para modificar el aspecto del portal web. Este es el único archivo requerido para hacer un tema hijo. La hoja de estilos debe empezar con las siguientes líneas:

```
/*
Theme Name: Twenty Fourteen Child
Theme URI: http://example.com/twenty-fourteen-child/
Description: Twenty Fourteen Child Theme
Author: John Doe
Author URI: http://example.com
Template: twentyfourteen
Version: 1.0.0
Tags: light, dark, two-columns, right-sidebar, responsive-layout, accessibility-ready
Text Domain: twenty-fourteen-child
*/

@import url("../twentyfourteen/style.css");

/* =Aquí empieza la personalización de tu tema
----- */
```

Las únicas líneas que se deben mantener son las de *Theme Name* y la de *Template*. *Template* es el nombre de la carpeta del tema padre, el resto son opcionales. En este caso, el tema padre es *Coller*, así que *Template* será *coller*, que es el nombre de la carpeta donde reside el tema *Coller* y el *Theme Name* es *childcoller*.

La hoja de estilos del tema hijo se incluye después de la del tema padre, por lo que los estilos del hijo prevalecerán sobre los del padre sobreescribiéndolos en los casos que sea necesario.

Para activar el nuevo tema creado hay que entrar en la administración de WordPress y en la sección de temas ya estará visible en el listado de temas como un tema más el que se ha creado. Presionar *activar* para utilizarlo. A partir de este momento cualquier cambio se haga en dentro de la hoja de estilos del tema hijo se verá afectado dentro del portal web.

Si se quiere cambiar algo más que la hoja de estilos, el tema hijo puede sobrescribir cualquier archivo del tema padre, simplemente incluye un archivo del mismo nombre en la carpeta del tema hijo y sobrescribirá el archivo equivalente en la carpeta del tema padre, excepción hecha con el archivo *functions.php* que a diferencia de los demás se carga justo antes del tema padre conviviendo ambos.

El archivo *functions.php* en WordPress es el archivo PHP que es utilizado dentro de WordPress para crear funciones que añadan o modifiquen comportamientos al portal web.

Se ha considerado oportuno utilizar el *functions.php* dentro del *child theme* para la creación de los *meta box* que añaden información adicional a los eventos y la conexión con Parse, para la modificación del estilo del portal web se ha creado también un archivo *style.css*.

3.1.3 Meta box

Un meta box permite añadir campos extra a la página de edición de las entradas o páginas. Con estos meta box se agrega más información sobre un mismo tipo de contenido.

En el sistema que se ha realizado para la creación de eventos como entradas de WordPress se han creado varios meta box para añadir información adicional correspondiente a los eventos creados.

Esta información complementaria es de gran importancia en el sistema ya que además de incluir más información a mostrar en el portal web esta misma información es guardada y enviada hacia Parse para que esté disponible y sirva como fuente de información a la aplicación móvil.

Como se ve en la imagen 3.2, se han creado varios campos que ayudan a definir la hora, fecha, tipo de evento e incluso añadir texto explicativo del evento creado que se rellena por parte de un editor a la hora de crear una nueva entrada para el sistema.

- Para la creación de la hora de comienzo se tienen dos campos que definen la hora y los minutos.
- La fecha está definida por los campos correspondientes al día, mes y año.
- La elección del tipo de evento es elegida a través de varios campos de tipo 'checkbox', pudiendo elegir uno o varios tipos para un mismo evento. Los tipos de eventos están clasificados en: deportes, fiestas populares, charlas y conferencias, conciertos musicales, espectáculos, infantil, exposiciones y varios.
- En el área para incluir texto llamado “Descripción del evento” da la opción al editor de contenidos a añadir información, esta información no se mostrará en el portal web sino que se guarda para ser mostrada en la aplicación móvil.

Fecha y Hora de los Eventos

Descripción del Evento

Hora de Comienzo 0

00

Fecha del evento 1

Enero 2014

Tipos de eventos:

Deportes:

Fiestas Populares:

Charlas y Conferencias:

Conciertos musicales:

Espectáculos:

Infantil:

Exposiciones:

Varios:

Figura 3.2 Imagen de los meta box creados en WordPress

- **Añadir un meta box**

Para añadir un meta box basta con añadir el siguiente código al *functions.php* del tema que se va a utilizar. En nuestro caso del child theme creado para la ocasión.

```
< ?php add_meta_box($id, $title, $callback, $page, $context, $priority, $callback_args ); ?>
```

En el que los parámetros significan:

- **\$id** es el ID html del meta box. Su importancia reside para poder darle estilo al meta box.
- **\$title** es el título del meta box. En nuestro caso es “Información adicional de los eventos”.
- **\$callback** es la función a la que llamamos para hacer uso del meta box creado.
- **\$page** indica donde queremos mostrar nuestro meta box. Ya sea en

páginas (`$page`) o en las entradas (`$post`). Usaremos `$post` para usarlo en las entradas.

- **\$context** con este parámetro indicame donde queremos mostrar el meta box. “normal” significará que se muestre bajo el editor de la entrada o página, “side” coloca el metabox a la barra lateral de la página de edición .
- **\$priority** informa a WordPress la prioridad con la que queremos que se nos muestre nuestro meta box. “high”, “default” o “low” coloca la caja del metabox arriba, en su posición natural o abajo del todo respectivamente.
- **\$callback_args** argumentos que serán pasados dentro de la función *callback*.

3.1.4 SEO

SEO es la abreviatura de *Search Engine Optimization*, optimización en motores de búsqueda. Los objetivos al realizar un portal web pueden variar según la temática del portal, hacia que público va dirigido, quien es el autor, etc., pero el objetivo común de todas los que se realizan es conseguir que las visitas sean en el mayor número posible. Con las técnicas de SEO se trata de conseguir la mayor visibilidad de un sitio web situándolo en las primeras posiciones de los resultados de los buscadores de Internet como Google, Yahoo o Bing.

El SEO se basa en adecuar una web a los algoritmos de indexación y ordenamiento de los buscadores. Estos algoritmos han ido evolucionando cambiando los factores prioritarios para una mejor indexación. La razón de estos cambio se hace para premiar a los portales web de contenido de calidad y originales contra sitios web con contenido duplicado o anticuado.

WordPress es considerado el mejor gestor de contenidos cuando se habla en términos de SEO, aún así es recomendable mejorarlas para obtener mejores resultados, para ello en el sitio web creado se ha utilizado el plugin *WordPress SEO by Yoast*.

- **WordPress SEO by Yoast**

Existe una gran cantidad de complementos dedicados al SEO para WordPress, los más populares son All in one SEO Pack, Ultimate SEO y SEO by Yoast. Cualquiera de los tres proporciona la ayuda necesaria para realizar los ajustes necesarios para adaptarse a los motores de búsqueda, principalmente Google ya que es el copa la mayoría de las búsquedas realizadas por los usuarios.

La razón principal por la que se eligió el plugin SEO by Yoast se debe a la ayuda que ofrece en el editor de entradas y páginas para saber de antemano si lo que se está creando se hace bien en términos de SEO.

La descarga, instalación y activación de este plugin se realiza como cualquier otro plugin por parte del administrador. Se puede hacer desde el escritorio de WordPress en la sección de plugins, o descargarlo desde la sección de plugins de la página oficial de WordPress o la web del plugin e incluyéndolo dentro del directorio plugins donde está instalado WordPress.

- **Apartados del plugin**

Un administrador del sistema también es el encargado de configurar correctamente las distintas opciones del plugin para hacer que funcione correctamente y tenga un buen posicionamiento en los buscadores.

Títulos y Etiquetas Meta.

Títulos y meta descripciones se refiere al título y la descripción de la página en la que nos encontramos. La etiqueta 'title' indica a Google el título de la página y además se muestra en las pestañas de los navegadores al situar el ratón sobre ellos. Los motores de búsqueda utilizan la meta descripción para describir la página en las páginas de resultados. Google recomienda una longitud de unos 155-160 caracteres. La figura 3.3 muestra cómo Google crea el listado de los resultados de una búsqueda a través de título y la meta descripción.



Figura 3.3 Títulos y Metas

Social

Las redes sociales cada vez van teniendo más peso a la hora del SEO en los sitios web, es por ellos que ofrece una opción para asociar el portal web a cuentas en redes sociales de Facebook, Twitter y Google+.

Sitemaps XML

Crear un Sitemap, mapa del sitio web, es muy importante ya que facilitamos a los motores de búsqueda la navegación por nuestro portal para indexarlo correctamente. Al activar el Sitemap el plugin lo actualiza automáticamente cada vez que se publican nuevas entradas o páginas.

Enlaces permanentes o permalinks

Este apartado permite configurar para que el portal web funcione indistintamente con www o sin www para que no exista duplicidad. En el caso de aceptar las dos opciones Google indexará hacia la marcada como principal.

Enlaces internos

Esta opción es importante de cara a la usabilidad del portal web debido a que si activamos los enlaces internos o *migas de pan* indicamos al usuario en qué lugar de la web se encuentra. Esta opción es recomendable para sitios web grandes con varios niveles de menú.

RSS

Esta función es útil a los motores de búsqueda a identificarte como el autor original de un contenido. Lo que hace es añadir contenido automático a tu RSS, concretamente un enlace de vuelta a tu web y a tus entradas, para que si alguien copia tu contenido automáticamente añada también el enlace a tu web y Google sepa de donde proviene la fuente original.

Éstos son los apartados más importantes del plugin, la configuración de ellos viene en función de las necesidades de cada desarrollador, por ese carácter subjetivo no se va a mostrar en detalle cual es la configuración del plugin. Lo que sí se va a detallar a continuación es la edición de una página o entrada de forma que esté optimizada para los buscadores según las indicaciones del plugin.

3.1.5 Configuración de una página o entrada

La configuración de las entradas se hará individualmente por los editores en el momento de crear una nueva entrada en la aplicación. En la sección de creación de una nueva entrada aparecerán los meta box correspondientes al plugin que se deberán configurar como se describe a continuación. La configuración de las páginas la realizará el administrador de la aplicación.

El editar una entrada o página en WordPress consiste en hacerlo en relación a una palabra clave.

En la pestaña General, puedes definir aspectos fundamentales para la optimización del portal web. Lo primero es seleccionar una palabra clave, al

insertarla nos realiza un análisis básico de algunos parámetros, como que esté presente en la metadescripción, en el contenido del artículo, título de la página, etc, y en base a esa información podemos ir corrigiendo para optimizar la página para los buscadores. Al guardar el contenido mostrará en rojo los parámetros que merecen ser mejorados y en verde lo que están correctamente configurados.

También nos brinda ayuda en relación a controlar la longitud del enlace permanente (*slug*), del Título SEO y Meta descripción (cuanto más larga sea, menos posibilidades tenemos de que sea indexada por los buscadores), el *plugin* nos indica cuando el número de caracteres es correcto o cuando lo hemos excedido. También nos muestra un vista preliminar del *Snippet* (que es como se mostrará esa página como resultado en un buscador). La figura 3.4 muestra a modo de ejemplo cómo es la pestaña general al configurar una página o una entrada.

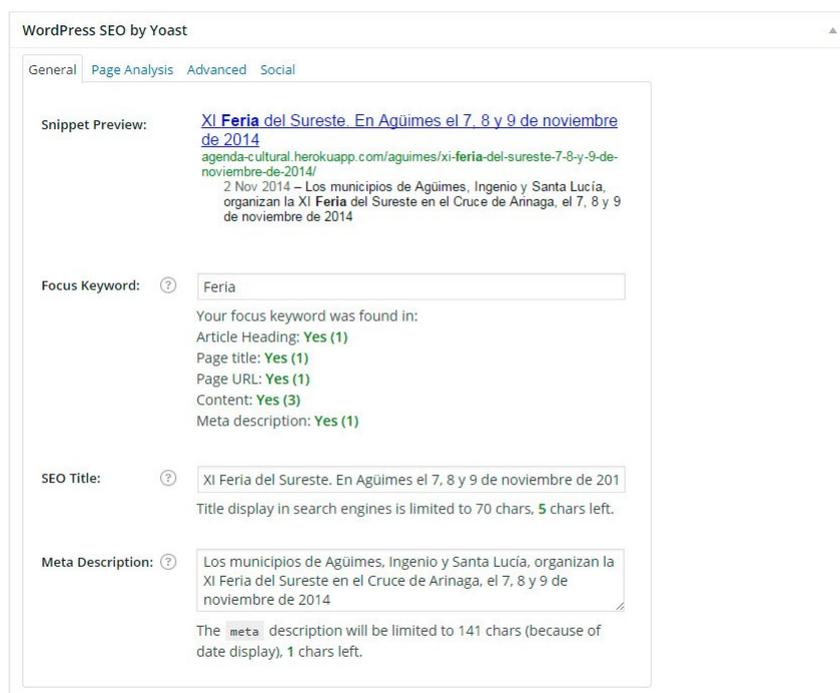


Figura 3.4 Análisis general de SEO

En la pestaña Análisis de página, presentada en la imagen 3.5, nos analiza muchos más parámetros, como pueden ser que tengamos imágenes en el texto que contengan en la etiqueta la palabra clave, enlaces externos, densidad de la

palabra clave, número total de palabras en la página, si la palabra clave está al principio o al final del título, si la palabra clave aparece en el primer párrafo, si tenemos encabezados que contenga la palabra clave, si hemos utilizado la palabra clave en otras entradas o páginas, etc.

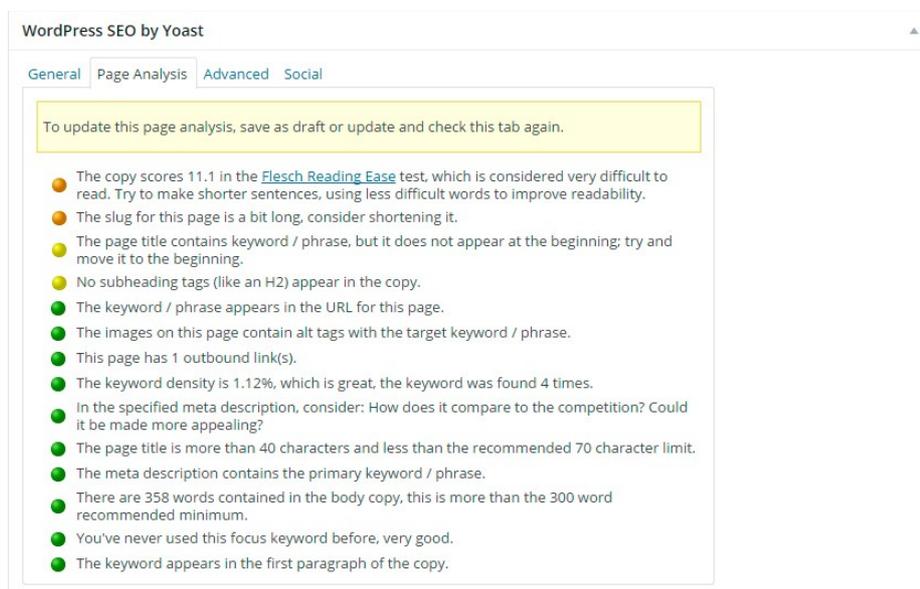


Figura 3.5 Análisis de la página de SEO

En la pestaña Avanzado, podemos controlar específicamente para la página temas de título de la ruta de las migas de pan, incluir en el Sitemap xml o html y su prioridad, añadir URL canónica.

Por último, en la pestaña Social, podemos especificar la metadescripción para Facebook y Google+.

3.1.6 Creación de entradas y páginas

- **Creación de una entrada**

La creación de una entrada es realizada para incorporar un nuevo evento cultural en el sistema para que sea visible en el portal web y en la aplicación móvil. Ésta es la operación principal que realizan los editores de contenido de la aplicación para la creación de los eventos e informar de las actividades culturales que se ofrecen en la isla de Gran Canaria.

Un editor, o administrador si se diera el caso aunque ya sabemos que no es su cometido principal, en el momento de crear un evento accede a la sección “*Añadir nueva entrada*” dentro del menú de WordPress. Una vez dentro se presenta algo similar a un editor de texto.

Se presentan dos campos para la introducción de texto, en el primero se añade el título del evento que se quiere crear y en la inferior se añade el cuerpo de la entrada que debe aparecer en la página del evento en el portal web. Para desarrollar el contenido de la entrada, WordPress facilita una barra de herramientas para darle forma al texto. Con estas herramientas puede darle forma a sus textos o introducir imágenes y vídeos. Para añadir imágenes, que es lo más habitual que se vaya a hacer ya que normalmente se incluirá el cartel del evento si lo hubiese, tenemos el botón *Añadir objeto* con el que seleccionaremos la imagen a incluir en la entrada. Una vez seleccionada la imagen, ésta se modificará a nuestras necesidades pudiéndola disminuir de tamaño o colocarla en cualquier parte del cuerpo de la entrada.

Otra de las posibilidades que ofrece WordPress con la creación de una entrada es la posibilidad de insertar código HTML, esta característica hace que las posibilidades para crear el cuerpo de las entradas se vea ampliada.

Esta característica que ofrece WordPress es utilizada en la creación de eventos de incluir código HTML es utilizada para aportar información a los usuarios sobre el lugar en el que se realiza un evento en concreto. Para incluir esta información en una entrada se realiza a través de la web de Google Maps, que ofrece la posibilidad de copiar la URL del lugar que se desee y adjuntarla como código HTML en una entrada de WordPress. Así en el página del evento se verá un recuadro con el punto exacto donde se desarrolla cada evento.

A la derecha están la categorías que se han incluido para clasificar los eventos, como se comentó las categorías son los municipios de Gran Canaria. Basta con seleccionar los municipios a los que se quiere agregar el evento que se está creando para que aparezca dentro del conjunto de eventos de ese municipio.

En la imagen 3.6 que se muestra en la siguiente página a modo de ejemplo, se realiza la creación de un evento, en él se selecciona el municipio de Agüimes ya que es allí donde se va a dar lugar, al publicar la entrada

automáticamente el evento quedará clasificado dentro de los eventos del municipio de Agüimes tanto dentro del portal web como en la aplicación móvil.

En la parte inferior del primeramente se encuentra el cuadro correspondiente al *plugin* de SEO, que se será rellenado como se ha detallado, poniendo énfasis en las palabras claves para que esté optimizada para la búsqueda de los buscadores.

Por último está el campo perteneciente a los meta box. En ellos se añaden un breve comentario sobre el evento y se seleccionan la fecha y la hora a la que va a tener lugar el evento. Para terminar se seleccionan la categoría o categorías a las que pertenece. En el caso de la Figura 3.6 que es mostrada a continuación se ve como la opción seleccionada es “varios”.

Una vez rellenados todos los campos del editor de entradas de WordPress, tanto los que vienen por defecto como los creados mediante los meta box, el evento es publicado y así ser visible tanto en el portal web como en la aplicación móvil. La acción de publicar un evento conlleva que esos datos sean enviados y almacenados en Parse para que sea posible su lectura, y así mostrarlos en los móviles.

La edición de entradas en WordPress ofrece la opción de borrar entradas por lo motivos que sean oportunos, para ello existe el botón papelera o *trash*. Esta acción hace que un evento no se muestre en el portal web.

Como las entradas en el sistema vienen asociadas a la creación de eventos se deduce que un evento puede ser borrado por diferentes motivos como puede ser que se hayan cometidos errores al crearlo o que finalmente el evento sea cancelado.

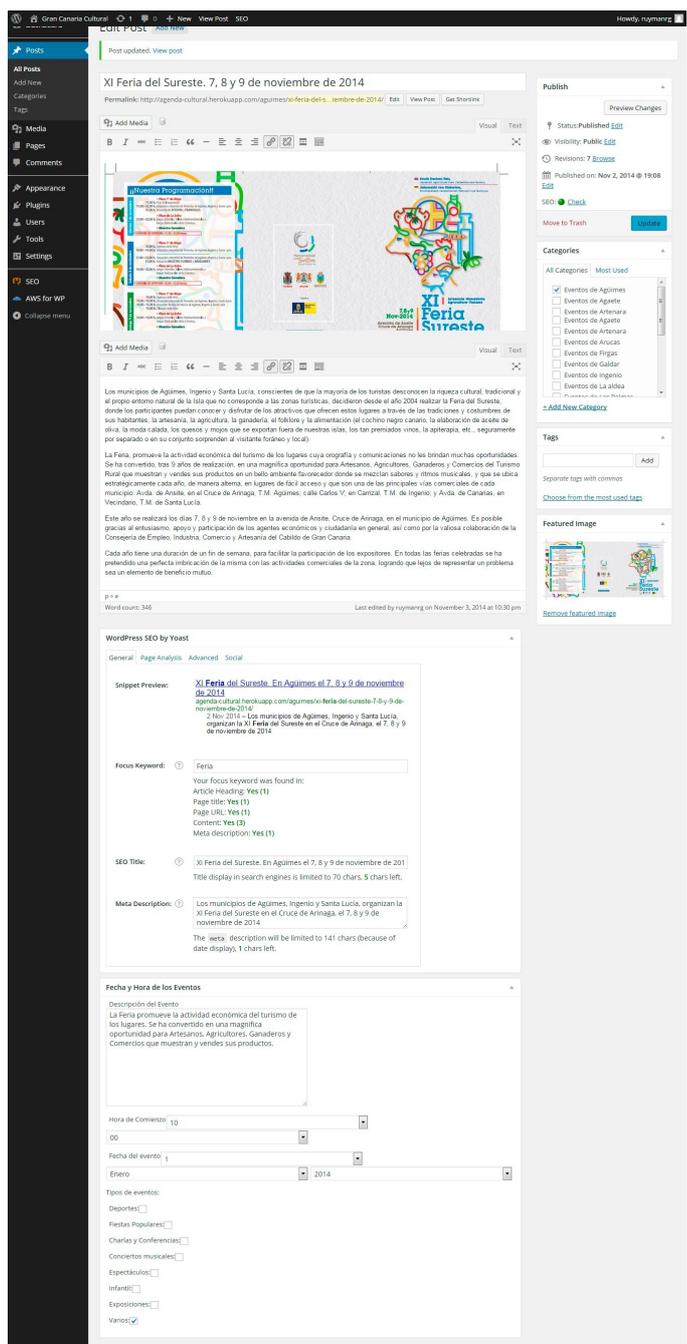


Figura 3.6 Crear evento

Como se ha comentado, la publicación de un evento conlleva el almacenamiento en Parse de los datos correspondientes al evento. Mediante una función creada para este proyecto, en el momento que un editor de contenidos borra un evento automáticamente los datos que fueron guardados en Parse son borrados, y por consiguiente en la aplicación móvil.

- **Creación de una plantilla de página**

Las plantillas de las páginas que vienen por defecto en un tema están en el directorio raíz de cada tema. Para crear una nueva plantilla de página se debe crear un nuevo archivo PHP. Supongamos que se va a crear la página agenda.php.

Para crear un archivo PHP se realizará desde el editor de código PHPStorm, donde se creará un nuevo archivo y será incluido en el directorio del tema de WordPress.

Las primeras líneas del archivo creado debe llevar el nombre de la plantilla, lo usual y aconsejado para evitar errores es llamar a la plantilla igual que se llamó al archivo, en este caso se llamará agenda. El siguiente código muestra cómo se declara el nombre de una plantilla.

```
<?php
/*
Template Name: Agenda
*/
?>
```

Incluyendo estas líneas de código ya se está en disposición de poder elegir esta plantilla dentro de los atributos de página junto con el resto que venía por defecto. Si se activara esta plantilla mostraría una página en blanco ya que no se ha incluido más código. El resto del código que se escriba en el archivo controlará como se mostrarán las páginas que usan la plantilla de página Agenda.

Para el proyecto se ha utilizado la creación de páginas para distinguir las diferentes partes en las que se divide el portal web. Éstas son agenda, calendario, usuario, registrarse y contacto.

3.2 Zend Framework 2

3.2.1 MVC

La arquitectura MVC (Modelo, Vista, Controlador) fue diseñada para reducir

el esfuerzo de programación necesario en la implementación de sistemas múltiples y sincronizados de los mismos datos. La característica principal de este patrón es que trata por separado el Modelo, la Vista y el Controlador, lo que permite que cada elemento sea tratado como una entidad separada, permitiendo que el mantenimiento del software sea de forma más clara, fácil y rápida para el programador.

Este patrón es ampliamente adoptado por los desarrolladores de aplicaciones web ya que permite una separación limpia entre los componentes para una mejor organización del trabajo y una mayor especialización de los desarrolladores y diseñadores.

Al incorporar el modelo de arquitectura MVC a un diseño, las piezas de un programa se pueden construir por separado y luego unirlos en tiempo de ejecución. Si uno de los componentes, posteriormente, se observa que funciona mal, puede reemplazarse sin que las otras piezas se vean afectadas.

Las principales ventajas que ofrece el patrón MVC son:

- La separación de los modelos en entidades separadas para facilitar su trabajo y la corrección en caso de fallos.
- Interfaz de Programación de Aplicaciones API (Application Programming Interface) muy bien definida; cualquiera que use el API, podrá reemplazar el Modelo, la Vista o el Controlador, sin gran dificultad.
- La conexión en el Modelo y sus Vistas es dinámica, se produce en tiempo de ejecución, no en tiempo de compilación.
- **Elementos de la arquitectura**
 - **El Modelo.** Es el que representa los datos del programa. Es el encargado del manejo de los datos y controla sus transformaciones. El Modelo no tiene conocimiento de las Vistas o del Controlador, ni siquiera contiene referencia a ellos. El Modelo es independiente del sistema de almacenamiento.
 - **La Vista.** Es el objeto que maneja la presentación visual de los datos

representados por el Modelo. Genera una representación visual del Modelo y muestra los datos al usuario. Interactúa preferentemente con el Controlador, pero es posible que trate directamente con el Modelo a través de una referencia al propio Modelo. En las aplicaciones web la Vista es la página HTML y el código que recoge los datos dinámicos para la página.

- **El Controlador.** Procesa y gestiona los eventos, por lo general las acciones del usuario. Interactúa con el Modelo y las Vistas por lo que puede invocar cambios en ambos.

A continuación se explicará el flujo que define el patrón MVC, aunque es posible encontrar varias implementaciones que difieran del original. La figura 3.7 se presenta a modo de ejemplo un resumen de los elementos y el flujo de los datos de la arquitectura MVC.

- El usuario interactúa con la interfaz de algún modo a través de la vista (por ejemplo pulsando un botón).
- Un controlador recibe y gestiona la acción que acaba de realizar el usuario en la interfaz, en este caso el pulsado del botón.
- Este controlador accede y modifica el modelo, normalmente actualizándolo, de acuerdo con la acción solicitada por el usuario.
- El controlador delega a los objetos de la vista la tarea de desplegar la interfaz de usuario. La vista obtiene sus datos del modelo para generar la interfaz apropiada para el usuario donde se refleja los cambios en el modelo. El modelo no tiene conocimiento directo sobre la vista.
- La interfaz de usuario espera nuevas interacciones del usuario, comenzando el ciclo nuevamente.

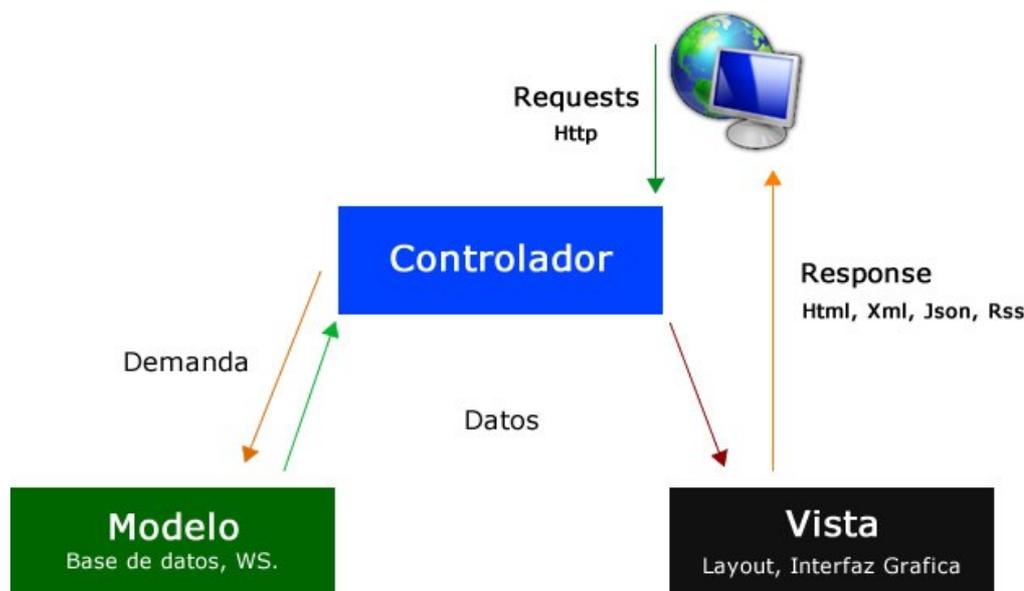


Figura 3.7 Esquema de la arquitectura MVC

3.2.1 Framework

Es una estructura de soporte definida mediante la cual otro proyecto de software puede ser organizado y desarrollado. Normalmente incluye soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto. Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio.

Su principal cometido es facilitar el desarrollo de software posibilitando a diseñadores y programadores concentrar más tiempo y esfuerzo identificando requerimientos que tratando con detalles de bajo nivel. Hay quejas comunes acerca de que el uso de *frameworks* añade código innecesario y que la preponderancia de *frameworks* competitivos y complementarios significa que el tiempo que se pasaba programando y diseñando ahora se gasta en aprender a usar *frameworks*.

Un *framework* no debe ser consciente de todos estos requerimientos sin tener que ser intrusivo con las aplicaciones que permite dentro de sí mismo, es decir, que actúa como armazón que sostiene los distintos objetos que se integran

para alcanzar una solución. A esto se le añade la capacidad de extenderse sin prejuicios para diversificar la expresión del programa mismo.

- **Frameworks MVC**

Los patrones de MVC son similares a los utilizados por los *frameworks* para desarrollar su trabajo, que no es otro que proveer al desarrollador de una estructura bien organizada que de soporte a un proyecto para que éste esté bien desarrollado y perfectamente organizado.

El uso de los *frameworks* basados en este patrón permite tener una separación lógica y física de los componentes de la aplicación, ya que por un lado se tienen los modelos, por otro las vistas y por otro los controladores. De esta forma, los desarrolladores de la aplicación pueden centrarse en la parte que les toca, ya sea como diseñadores en las vistas, o como programadores de los modelos del negocio.

3.2.3 MVC en aplicaciones web

El patrón MVC se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página; el modelo es el Sistema de Gestión de Bases de Datos y la Lógica de negocio; y el controlador es el responsable de recibir los eventos de entrada desde la vista.

Utilizar el patrón MVC provee grandes beneficios a la programación Web, aparte del ya indicado de la organización del proyecto.

- Muchos *frameworks* MVC incluyen librerías de Javascript como JQuery, lo que te facilitará validar formularios.
- Si se trabaja en un grupo de trabajo, seguir un método común de programación hace que el código sea más entendible para todos, pudiendo continuar el trabajo del otro. Para ello es imprescindible utilizar herramientas de control de versiones como Bitbucket o Github.
- La mayoría de *frameworks* controlan el uso de la memoria caché, hoy en

día muy importante para el posicionamiento web, ya que buscadores como google dan prioridad a las webs que tengan menor tiempo de descarga.

- Facilita el uso de URL amigables, importantes para el SEO, la mayoría de *frameworks* MVC lo controlan.

Para la creación del portal web se ha usado el *framework* Zend Framework

2.

3.2.4 Zend Framework 2

Zend Framework 2, comúnmente conocido como ZF2, es la segunda versión de Zend Framework desarrollado por la empresa Zend Technologies en Septiembre de 2012, es un *framework* de código abierto para el desarrollo de aplicaciones webs. ZF2 es una implementación que usa código 100% orientado a objetos. La estructura de los componentes de Zend Framework es algo único, cada componente está construido con una baja dependencia de otros componentes. Esta arquitectura débilmente acoplada permite a los desarrolladores utilizar los componentes por separado.

ZF2 ofrece un gran rendimiento y una robusta implementación MVC, una abstracción de base de datos fácil de usar, y un componente de formularios que implementa la prestación de formularios HTML5, validación y filtrado para que los desarrolladores puedan consolidar todas las operaciones usando de una manera sencilla la interfaz orientada a objetos. También provee de autenticación y autorización de usuarios.

ZF2 está diseñado con el objetivo de simplificar el desarrollo de sistemas web implementando las mejores prácticas y patrones de diseños de ingeniería de software, enfocado en la construcción de seguras, fiables y modernas aplicaciones Web 2.0.

Zend Framework 2 ofrece al usuario las siguientes capacidades:

- Desarrollo de un portal web de forma más rápida que con la utilización de código PHP puro. Debido en gran parte a la gran reutilización que hace del código permitiendo volver a utilizar modelos, vistas y controladores de un

sitio web en otros trabajos.

- Facilita la cooperación para trabajo en equipo. Al utilizar el patrón MVC separa la lógica de negocio con la capa de presentación del sitio web, por lo que su estructura es consistente y de fácil mantenimiento.
- Acceso a la base de datos de una manera orientada a objetos, no realiza la consulta directamente mediante consultas SQL.
- Garantiza la creación de portales web seguros proporcionando validadores, filtros y algoritmos de criptografía (Captcha) a los elementos de los formularios.

Sin duda es el *framework* más apropiado para trabajar con PHP ya que está desarrollado por los mismo creadores de PHP, por lo que son los que más se aproximan al estándar. En la actualidad hay grandes compañías como la BBC (Corporación Británica de Radiodifusión) o BNP Paribas (Uno de los mayores bancos de Francia y Europa) que se decantan por la utilización de Zend Framework para la creación de sus portales web.

- **Estructura de un proyecto en ZF2**

La estructura de un proyecto en ZF2 se divide según se explica a continuación:

- **config**: Se utiliza para la gestión de configuración de la aplicación.
- **data**: Carpeta de almacenamiento temporal para almacenar datos de la aplicación como archivos caché o de sesión.
- **module**: Directorio donde se gestiona todo el código de la aplicación.
- **public**: Aquí se incluyen todos los recursos de la web, archivos CSS y Javascript, así como las imágenes.
- **vendor**: Permite gestionar las bibliotecas comunes utilizadas por las aplicaciones.
- **vendor/zendframework**: Zend Framework está instalado en esta carpeta.

En ZF2 los proyectos se dividen en módulos. Un módulo se define como una unidad de software que es portátil y reutilizable y puede ser interconectada con otros módulos para construir una aplicación compleja más grande. El módulo por defecto que crea ZF2 se llama *Application* como se expone al final de esta página en la figura 3.8, y como se comentó es donde se incluye todo el código que se creará.

Cada módulo contendrá, siguiendo el patrón MVC, su propio conjunto de modelos, vistas y controladores, que puede compartir con otros módulos y se denominan en ZF2 como config, src y view.

- Config: Se utiliza para la gestión de configuración del módulo.
- Src: Contiene todo el código fuente del módulo, incluyendo todos los controladores y modelos.
- View: Se utiliza para almacenar todas las vistas usadas en el módulo.

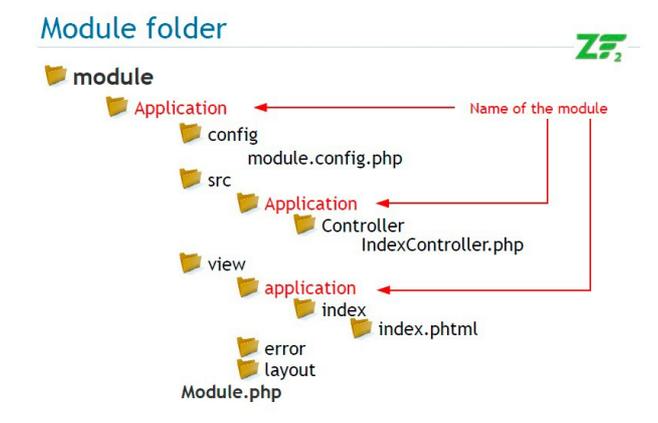


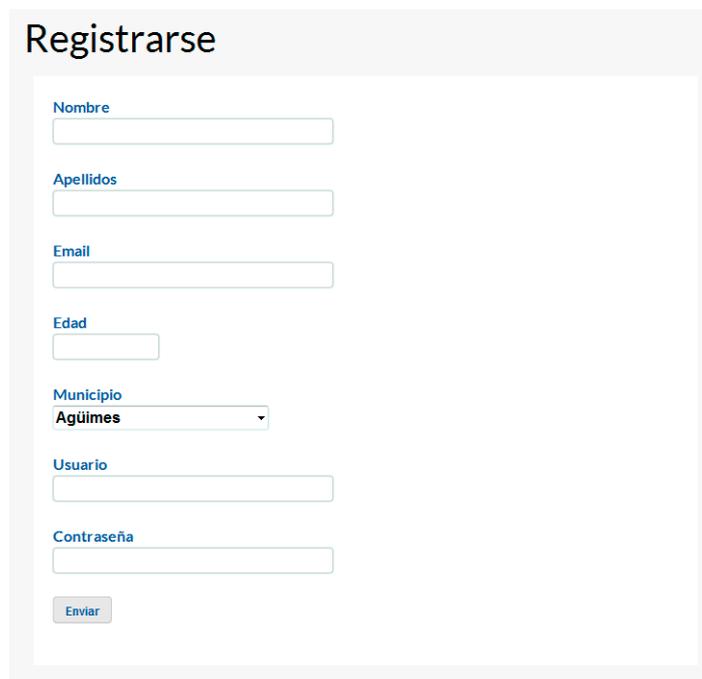
Figura 3.8 Estructura de un módulo de ZF2

3.2.5 Zend Framework 2 en el sistema

Como se ha comentado, Zend Framework 2 se ha utilizado para la creación de los formularios que se incluyen en la aplicación. Por un lado está el formulario de registro de usuarios y por otro el formulario de contacto con la administración del sistema, ambos representados en las figura 3.9 y 3.10.

ZF2 al utilizar el patrón MVC es ideal para la reutilización de código. Se crearon todos los campos en un archivo PHP y a continuación se crearon sendas vistas que llamaban a los campos necesarios para cada formulario.

ZF2 se integró en WordPress a través del plugin Wopzen2. Cuando un usuario entraba en el sistema y rellenaba alguno de los formularios WordPress obtenía los valores ingresados y los enviaba a Parse para su almacenamiento si se trataba del formulario del registro de usuarios o para enviarlos por correo si se trataba del formulario de contacto.



Registrarse

Nombre

Apellidos

Email

Edad

Municipio
Agüimes

Usuario

Contraseña

Enviar

Figura 3.9 Formulario para el registro de usuarios



Contacto

Nombre

Apellidos

Email

Introduce texto

Enviar

Figura 3.10 Formulario de contacto con al administración del sistema

Servicios en la nube

Decididos los servicios en la nube que forman parte del proyecto y la función que desempeñan cada uno de ellos, se procede en este capítulo a realizar una descripción más detallada y amplia de cada uno de ellos y la función que realiza cada uno de estos servicios dentro del sistema.

4.1 Heroku

Heroku es una plataforma como servicio (Paas) que ofrece el alojamiento de aplicaciones en la nube. Heroku es un servicio de publicación de aplicaciones nacido en el 2007 y que inicialmente fue creado para ejecutar aplicaciones programadas en Ruby on Rails, pero con el éxito de su servicio se amplían su funcionalidad y pasando a soportar múltiples lenguajes y plataformas.

Esta plataforma es la plataforma como servicio elegida en el proyecto para instalar y desplegar la aplicación web, es decir, Heroku es la plataforma donde se realiza la instalación de WordPress.

Actualmente Heroku es una plataforma que permite ejecutar y administrar una aplicación escrita en varios lenguajes como Java o PHP. Para que una aplicación pueda ser instalada en Heroku debe haber un *buildpack* disponible para ese tipo de lenguaje. Los *buildpack* son un conjunto de instrucciones que permiten ejecutar una aplicación en Heroku. Al existir uno para lenguaje PHP es lo que permite hacer una instalación de WordPress.

Los requisitos para crear una aplicación basada en PHP en Heroku son los tres siguientes:

- Tener conocimientos de PHP
- Estar dado de alta en la plataforma Heroku.
- Instalar Heroku Toolbet. Herramienta que ofrece Heroku para gestionar sus aplicaciones y que permite subir las aplicaciones a la plataforma.
- Tener instalado el sistema de control de versiones Git.

- Tener instalado *composer.json*. *Composer* es un organizador de dependencias de PHP que maneja las librerías que se tienen que instalar para que un proyecto de PHP funcione correctamente.

Heroku ofrece la opción a los usuarios de la plataforma un plan gratuito con el que realizar aplicaciones web de un tamaño reducido. Aunque una de las ventajas comentadas de Heroku, como de la mayoría de servicios en la nube, es la escalabilidad, que permite ir aumentando los recursos según las necesidades del cliente si la web necesita de más recursos.

La versión gratuita de Heroku no almacena gran cantidad de datos debido a que no tiene servidores propios, sino que los contrata de la nube EC2 de Amazon y sólo provee de una pequeña cantidad para alojar la aplicación. El almacenamiento que ofrece Heroku en su versión gratuita para la instalación de aplicaciones es de 300MB. Por esta razón, si es necesario el almacenamiento de archivos para el correcto funcionamiento de la aplicación creada, hay que buscar otro servicio en la nube donde almacenar todos estos archivos. Para el almacenamiento de archivos en este PFC se utiliza el servicio de un CDN donde se guarda el contenido estático y dinámico generado en la web.

Para trabajar con esta plataforma se hace a través de la consola usando el sistema de control de versiones Git. Para instalar la aplicación o hacer cualquier modificación se hace un *push* del código hacia el repositorio creado en Heroku. Hacer un *push* significa subir el código creado o modificado hacia la plataforma. Al realizar el *push* automáticamente se reiniciará la aplicación con la nueva versión instalada.

Una de las limitaciones que ofrece Heroku para trabajar con WordPress viene dada porque su versión gratuita sólo soporta el sistema de gestión de bases de datos PostgreSQL, y no MySQL. Esta limitación es un inconveniente para los desarrolladores que crean portales web con el CMS de WordPress ya que WordPress trabaja en su base de datos con MySQL. Existe una solución para trabajar con la base de datos PostgreSQL en WordPress, a través del uso de un plugin, pero esta solución limita la capacidad de creación del portal web ya que varios plugins no trabajan correctamente en una base de datos PostgreSQL.

4.1.1 Dynos

El elemento más importante dentro de la arquitectura de Heroku es el *dyno*. Cada *dyno* es una maquina virtual de UNIX que ejecuta un único comando especificado por el usuario. En clave de aplicación web, un *dyno* es un servidor web que acepta peticiones HTTP de los usuarios. Para obtener más recursos en Heroku es a través de la contratación de un mayor número de *dynos*. La capa gratuita de Heroku permite crear a un usuario todas las aplicaciones que quiera, pero siempre asignando un *dyno* por aplicación.

Cada *dyno* tiene asignado un máximo de 512MB de memoria. Esta capacidad es suficiente para la gran mayoría de las aplicaciones por lo que el desarrollador no debe preocuparse por asignar más memoria a su aplicación.

Una de las desventajas de la plataforma en su versión gratuita, en la que sólo se tiene un *dyno* en ejecución, es que si una aplicación web no recibe peticiones HTTP en un tiempo superior una hora, el único *dyno* que tiene asignado la aplicación entra en el estado de “dormido” y deja de estar en ejecución. Al pasar a ese estado cuando la aplicación recibe la primera petición HTTP hace que el *dyno* “despierte” comenzando a ejecutar el proceso web. Este proceso de pasar de pasar de dormido a despierto hace que la primera petición tenga una demora de unos pocos segundos en la respuesta . Ésto hace que la carga de la web sea lenta para el usuario que realiza la acción de despertar al *dyno*. Las siguientes peticiones que se hagan tendrá una respuesta y rendimiento normal.

Heroku automáticamente proporciona a cada aplicación 750 horas de *dynos* al mes. Esto significa que una aplicación que solo tiene un web *dyno* consume 720 horas cada mes, por lo que puede funcionar sin coste de forma indefinida.

4.1.2 Modificación de la aplicación en Heroku

Para la modificación del código instalado en una aplicación de Heroku se realiza a través del sistema de gestión de versiones git. En el caso de la

instalación creada para el proyecto la modificación de la aplicación se ciñe a la modificación de los elementos de WordPress. Estas acciones son la de añadir, eliminar y modificar código PHP o elementos de WordPress como temas o plugins. Para ello hay que acceder dentro del directorio donde está la aplicación en el ordenador y se realizan 3 operaciones para incluir las modificaciones en Heroku.

- **git add -all**. Este comando añade al índice cualquier fichero nuevo o algún archivo antiguo que haya sido modificado.
- **git commit -m "nombre-del-commit"**. Se hace un *commit* con los cambios realizados.
- **git push heroku production:master**. Se suben los cambios a la plataforma Heroku. Con esta acción se reinicia la aplicación con los cambios realizados.

4.2 CDN

Una red CDN (Content Distribution Networks) o red de entrega de contenidos, es un sistema de servidores conectados en red en el que existe un servidor central al cual están conectados servidores secundarios que contienen copias de los datos con el fin de que el usuario acceda a éstos de la forma más rápida posible.

Un CDN se encarga de replicar en sus distintos servidores distribuidos en diferentes áreas geográficas la información original, como ejemplo la figura 4.1 nuestra un CDN como Amazon tiene servidores alrededor del mundo.

Los dos objetivos principales de un CDN son:

- Equilibrar la carga entre los servidores evitando así los cuellos de botella.
- Disminuir el retardo, ya que en teoría cuanto más cerca esté el contenido y sea más rápida su entrega, será menor el tráfico de paquetes en la red.

Estas operaciones se hacen de forma transparente al usuario final por lo

que éste no notará diferencia al recibir los datos de uno u otro servidor. Por ejemplo, un usuario que accede a nuestra web desde Miami y descargue un fichero lo estará descargando directamente del Data Center en Estados Unidos en lugar de acceder a los servidores de Europa.

Inicialmente una red CDN se encargaba de guardar el contenido estático de una página web (imágenes, documentos, etc.) pero esto ha cambiado ya que se ha ido orientando su uso hacia la comunicación en tiempo real, sobre todo al video en *streaming*.

Existen diversas empresas que se encargan de la distribución de contenido, siendo las más conocidas Akamai Technologies y el servicio CloudFront de Amazon Web Services, pero hay una gran oferta en este terreno. Una de las ventajas que ofrecen las empresas CDN es relación calidad-precio ya que ofrece gran cantidad de almacenamiento y una velocidad elevada a un precio realmente bajo.

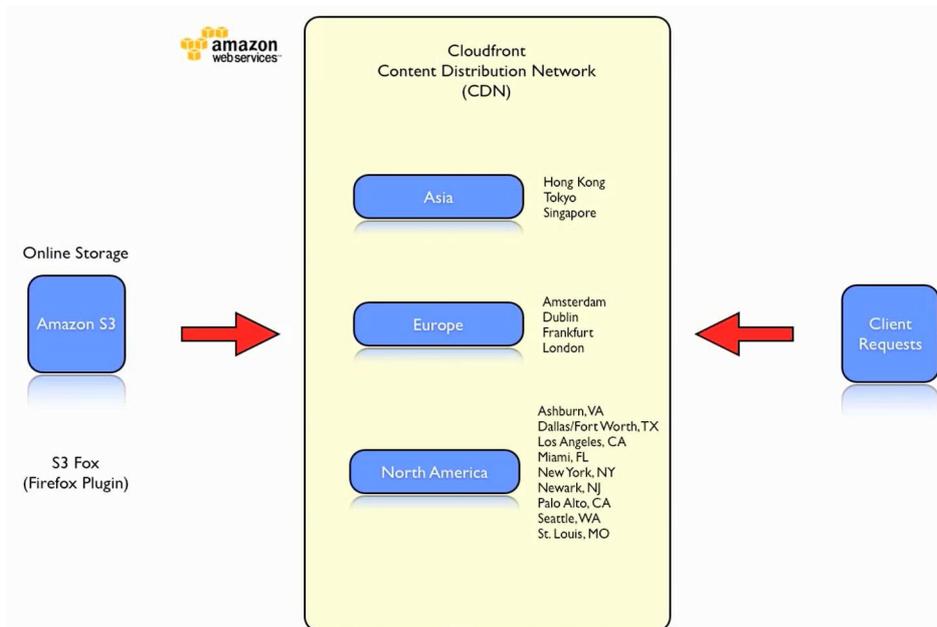


Figura 4.1 Esquema de funcionamiento del CDN Amazon Web Service

Muchas de las webs más importantes a nivel mundial utilizan un servicio CDN para la entrega de sus contenidos, entre ellas se pueden destacar la web de la NASA, la televisión pública de Estados Unidos (PBS) o Imdb, la web con mayor contenido mundial sobre cine.

Para el proyecto fin de carrera, para la acción de entrega del contenido estático manejado por el portal web se ha decidido hacer uso de un servicio CDN. Junto con la red de CDN es necesario un servicio de almacenamiento en la nube, que es el encargado de almacenar el contenido y hacer de entrega de él en el momento que sea requerido.

Como proveedor de CDN se ha utilizado el servicio CloudFront de Amazon Web Service que en conexión con el servicio Amazon S3, también de Amazon, realiza el servicio de almacenado de contenido.

El contenido que es almacenado en estos servicios en la nube está principalmente relacionado con la imágenes que se manejan en la creación de eventos. Además, se hará uso de él para almacenar distintas imágenes que ayudan a crear una mejor interfaz en el portal web y en la aplicación web.

Se ha utilizado como proveedor a Amazon Web Services para el almacenamiento de imágenes de nuestro sistema por varios razones:

- Ser un servicio de muy bajo coste y sin compromiso mensual en el que sólo se paga por el contenido que distribuya a través del servicio de entrega.
- Tener fama de ser un servicio muy eficaz ya que está avalado por grandes compañías que la utilizan.
- Ser utilizado por otros compañeros de proyectos anteriores que han realizado el proyecto fin de carrera con este mismo grupo de trabajo.
- La existencia de plugins que facilitan la integración con Heroku y WordPress.

4.2.1 Amazon Web Service

La empresa Amazon, con amplia experiencia en escalabilidad de entornos web proporciona una amplia selección de productos utilizando una infraestructura basada en servicios web. Amazon Web Service (AWS) representa la Infraestructura como servicio (IaaS) más grande actualmente. Algunos de los sitios web más importantes se están ejecutando sobre la infraestructura de

Amazon.

AWS tiene una política que hace que sea la más atractiva del mercado: se paga por lo que se usa. Ésto unido a unos precios realmente económicos hace que sus servicios sean los escogidos por la mayoría a la hora de elegir una infraestructura en la nube.

Amazon ofrece una larga lista de componentes entre los que destaca Amazon Elastic Compute (EC2). EC2 es una plataforma de servidor virtual que permite a los usuarios crear y ejecutar máquinas virtuales en los servidores de Amazon. El servicio EC2 es la Infraestructura como servicio de Amazon Web Service para la creación de aplicaciones en la nube.

Pero en los servicios en los que se va a centrar este proyecto son Amazon Simple Storage Service y Amazon CloudFront ya que son los utilizados para el almacenado y entrega de datos en nuestro sitio web.

Amazon S3

Amazon S3 (Simple Storage Service) proporciona almacenamiento de archivos en Internet. El almacenamiento y recuperación de esos datos se puede realizar al instante desde cualquier punto únicamente teniendo una conexión a Internet. Las características principales que ofrece este servicio son:

- **Seguridad.** Permite al usuario tener el control absoluto sobre quien puede tener acceso a los datos
- **Fiabilidad.** Los datos son almacenados con una fiabilidad del 99,999999999% y una disponibilidad del 99,99%.
- **Escalabilidad.** Ofrece escalabilidad respecto a almacenamiento, usuarios o velocidad de solicitudes.
- **Rapidez.** Ofrece una latencia insignificante respecto a la latencia de Internet. Corrección inmediata de los cuellos de botella.

Se cobra una tarifa mensual por almacenamiento y transferencia de datos que se ajusta al uso que se ha realizado de la infraestructura de almacenamiento

de Amazon durante ese mes, siendo el primer año de uso totalmente gratuito. Esta cualidad ayuda a comenzar un nuevo proyecto sin la necesidad de una gran desembolso de inicio para los desarrolladores. El almacenamiento en Amazon ofrece escalabilidad y simplicidad a bajo coste, resultando una opción adecuada para delegar la gestión de archivos multimedia (imágenes, vídeos, etc.) de cualquier servicio web.

Amazon CloudFront

Amazon CloudFront es otro de los servicios web que ofrece Amazon a terceros. Es una red de distribución de contenidos, también llamada computación en el margen, que sirve información desde diecisiete áreas geográficas alrededor del mundo actualmente. Amazon CloudFront se puede utilizar para entregar todo el contenido del sitio web, incluido el contenido interactivo, dinámico y estático, mediante la utilización de una red global de ubicaciones de borde. Las solicitudes de contenido se redirigen de forma automática hasta la ubicación más cercana, para que el contenido se entregue con el máximo rendimiento posible.

Creación de un CDN de Amazon para la web

Para la creación de un servicio de CDN en Amazon Web Service que sea compatible con WordPress para el almacenamiento de archivos es necesario la realización de tres pasos:

- Crear un carpeta de almacenamiento donde se guardará el contenido generado en el portal web.
- Establecer un servicio caché de la imágenes almacenadas.
- Instalación y configuración de un plugin que enlace WordPress con los servicios creados en Amazon.

- **Creación de un Bucket en S3.**

Una vez dados de alta, en la consola de AWS se accede a la sección S3 a través de la consola de la web de Amazon. Una vez en esta sección se crea un *bucket*. Un bucket o contenedor es una carpeta donde se guardarán todos los archivos que se suban desde el portal web. En el caso de la agenda cultural, en la gran mayoría serán las imágenes que acompañen a la creación de los eventos o imágenes que ayuden a crear la interfaz de usuario tanto en la web como en la aplicación móvil.

La parte más importante de la configuración es modificar los permisos del bucket para dar permiso de lectura para que pueda ser accesible a todo el mundo, es decir, a todos los usuarios que accedan al sistema y quieran visualizar las imágenes.

- **Creación del Servicio de Amazon CloudFront.**

La configuración del CloudFront se realiza creando lo que ellos llaman *Distribution* y completando la información entre la cual tenemos que elegir el bucket creado anteriormente para que Amazon realice la conexión entre el servicio CloudFront creado y el *bucket* establecido anteriorme y guarde allí todos los archivos subidos desde el gestor de contenidos.

- **Conexión con plugin de WordPress.**

Como se explica en la sección dedicada a los plugins de WordPress, para el proyecto se ha utilizado el plugin *Amazon Web Services for WordPress* que conecta el portal de Wordpress con el servicio de Amazon para el envío y almacenado de archivos.

Para configurar el plugin lo primero que hay que obtener son las claves que proporciona Amazon al crear una cuenta. Estas claves son *Access Key ID* y *Secret Acces Key*, que son la clave y la clave secreta. Estas claves hay que introducirlas dentro del panel de administración del plugin para indicar el usuario

al que se va conectar.

Una vez indicadas las claves hay que seleccionar el bucket al que quieres que se conecte el plugin para que almacene el contenido generado en la instalación de WordPress.

Rellenados estos campos, figura 4.2, ya se está en disposición de guardar todas las imágenes generadas en WordPress dentro del servicio S3 de Amazon y replicadas en los distintos servidores que Amazon tiene alrededor del mundo, y accesibles a todos los usuarios del sistema web y móvil.

La conexión de WordPress con el servicio de almacenamiento de Amazon se hace de forma automática y transparente para los editores de contenido. Un editor crea una nueva entrada en WordPress para dar de alta un nuevo evento en el sistema. Al seleccionar una imagen para añadirla a la nueva entrada, en el momento de subir esta imagen, automáticamente se conecta con el servicio de almacenado de Amazon que guardará la imagen en el *bucket* creado para nuestro sistema.

The screenshot shows the 'AWS For Wordpress - Amazon S3 and CloudFront' configuration interface. It is divided into several sections:

- AWS Access Credentials:** Includes a note about signing up for an Amazon S3 account and a link to 'sign up'. Below this, there are two input fields: 'Access Key ID' with the value 'AKIAJNSSZRW7JOLQEMMA' and 'Secret Access Key' with the value '-- not shown --'.
- S3 Bucket:** Features a dropdown menu labeled 'Choose an S3 bucket:' with the option '-- Select an S3 Bucket --'. Below the dropdown is a checkbox that is currently unchecked, with the text 'Uncheck this to revert back to using your own web host for storage and delivery at anytime.'
- CloudFront Settings:** Contains a 'Domain Name:' input field and a note: 'Leave this blank if you aren't using CloudFront.'
- Enable storage on S3/CloudFront:** Includes a checkbox that is unchecked, with the text 'Uncheck this to revert back to using your own web host for storage and delivery at anytime.'

At the bottom of the form is a blue button labeled 'Save Changes'.

Figura 4.2 Administración de Amazon Web Services for WordPress

Una vez que la imagen esté almacenada ya será accesible para todos los usuarios que accedan al sistema y por los administradores del sistema para hacer uso de ella.

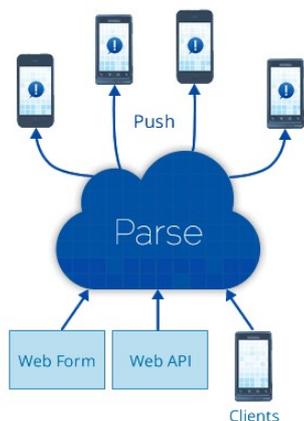
4.3 Parse

Parse es un potente servicio de gestión de base de datos en la nube que puede ser utilizado para el desarrollo de aplicaciones web y móviles. Destaca por ser uno de los servicios más fáciles y más rápidos de back-end para trabajar, así como tener la mejor documentación.

Parse es el líder en el sector de Baas con una cuota de mercado de un 28%, siendo el el segundo CloudMine con un 11%. Se estima que hay más de 100.000 aplicaciones que utilizan Parse.

Los servicios que ofrece Parse principalmente son:

- Modelo de datos en la nube, creación de tablas no-SQL en la nube y capacidades para inserción, modificación y consulta vía API.
- Notificaciones Push, posibilidad de envío de notificaciones push a nuestros usuarios, previa aceptación por parte del usuario.
- Cloud Code, capacidades para la ejecución de código en el servidor, muy útil para la realización de validaciones de seguridad, o procesos automáticos por cambios en los datos.



Esquema de funcionamiento de Parse

Parse es una aplicación tanto de pago como gratuita, siendo ésta última con unas dimensiones bastante grandes para iniciarse con las aplicaciones. La capa gratuita ofrece las siguientes funcionalidades al mes:

- 1.000.000 peticiones API.
- 1.000.000 notificaciones push.
- 1 Gb de espacio en disco.
- Un máximo de 30 peticiones por segundo.

La elección de Parse como BaaS para el trabajo se deben a tres razones principales:

- Su plan gratuito se adapta a las necesidades del sistema.
- Ser el Baas más utilizado del mercado entre los desarrolladores.
- Experiencia previa satisfactoria por parte del grupo de trabajo con el que desarrollo el sistema.

Utilización de Parse como base de datos

La elección de Parse en este proyecto fin de carrera se ha hecho con el fin de utilizarlo como base de datos de una parte de la aplicación. Como se comentó en el apartado dedicado a la base de datos y sus tablas, se han creado tres tablas donde guardar los datos que se refieren a los usuarios y a los eventos.

La tabla de Usuarios recoge los datos del formulario creado por Zend Framework 2 para el registro de usuarios. En el momento de registrarse, los datos del nuevo usuario son enviados a la tabla Usuarios de Parse para almacenarlos. Cuando un usuario registrado se identifica en el sistema éste accede a la tabla Usuarios para buscar si ese usuario está dado de alta en el sistema o no.

Las otras dos tablas han sido nombradas como *Categories* y *Products*. La primera de estas tablas ayuda a crear la estructura para la realización de la aplicación móvil mientras que la segunda, la tabla Products, almacena los datos

de los eventos creados en WordPress para que sean leídos por el framework creado por el IUMA y crear la aplicación. Estas dos tablas serán explicadas con más detalle en el capítulo 6 de la memoria.

4.3.1 Trabajando con Parse

Para comenzar a trabajar con Parse hay que acceder a su portal web <https://www.parse.com/> y darse de alta mediante un sencillo formulario como lo hay en multitud de portales web.

Dados de alta ya en la web en primer lugar deberemos crear en nuestra cuenta una nueva aplicación para poder comenzar a disfrutar de Parse. La aplicación creada en Parse es la llamada AgendaCultural. En esta aplicación se crean las tablas encargadas de almacenar los datos provenientes del portal web.

- **Crear tablas**

Para crear una tabla o Class que es como se denomina en Parse se accede en el panel a la pestaña Data Browser y creamos nuestra tabla pulsando New Class. Para guardar los datos generados del sistema se procede a la creación de tres tablas:

- Una tabla denominada *Usuarios* que almacena los campos de los usuarios dados de alta en el sistema desde el portal web.
- Dos tablas denominadas *Categories* y *Products*. En estas tablas se almacenan datos que serán de utilidad para la creación de la aplicación móvil. Se describirá con más detalle en el capítulo 6 de la memoria qué datos se almacenan en estas dos tablas de Parse.

- **Introducir campos a la tabla**

La creación de campos en una tabla se realiza añadiendo columnas a la clase. Cada columna creada será un campo de la tabla que almacena los datos del sistema.

La tabla Usuarios creada para guardar los datos de los usuarios dados de alta en el sistema tendrá tantos campos como cada uno de los campos del formulario *Registrarse* del portal web. Por lo tanto la tabla Usuarios tendrá los siguientes campos en relación con el formulario mostrado en el portal web.

Tabla Usuarios	Formulario web
nombre	nombre
apellidos	apellidos
edad	edad
email	email
municipio	municipio
usuario	usuario
pass	contraseña

Tabla 4.1 Relación Parse-Formulario

Diseño y estructura del portal web

Para plantear el diseño de un portal web es primordial tener varios puntos claros antes de emprender el trabajo de realizar la web.

- Por qué y para qué se crea el proyecto.
- Hacia qué tipo de usuarios va dirigida.
- Qué diseño de la interfaz de usuario se va a crear.
- Qué contenido se va a incluir en cada una de las partes en las que se divide la web.

Para el estudio del diseño, estructura y contenidos del sitio web se ha realizado un estudio más en profundidad siguiendo las pautas generales que marcan la creación de sitios web.

5.1 Diseño web

En el momento de plantearse el diseño de un sitio web tienen que convivir dos aspectos que determinan si un diseño web es bueno o malo. Por un lado está la usabilidad, que se centra en la funcionalidad y eficiencia del portal web y por otra la estética, que trata la forma de representar los elementos en una página web de la forma más atractiva para el usuario. Tiene que haber una compenetración de ambos para que al usuario le resulte atractivo a simple vista el diseño de una página web pero a su vez el contenido sea fácil de usar por el usuario medio del portal web.

Uno de los puntos principales de un diseño es que el usuario pueda moverse fácilmente por la web de forma intuitiva. El usuario debe navegar por un portal web casi sin la necesidad de pararse a pensar cuál es el próximo paso a realizar para seguir obteniendo información.

A través de estudios de usabilidad se ha llegado a trazar unas pautas sobre cual es la forma en la que los usuarios hacen uso de un portal web.[5]

- Los usuarios hojean las páginas. Un usuario al entrar en una página web no lee minuciosamente el contenido, lo lee por encima buscando algún contenido interesante o uno en concreto. Esta manera de actuar se acentúa cuando un usuario es un visitante nuevo en el portal web y busca una información determinada.
- Se toman rápidas decisiones. Se accede al primer enlace que se cree puede dar información sobre lo que se está buscando. Si la elección no es correcta siempre se puede volver hacia atrás a seguir buscando.
- Los usuarios no se interesan en entender el funcionamiento. Se salta de una página a otra intentando solucionar el problema antes de parar y pensar el lógico funcionamiento y estructura del portal web.

Teniendo datos de cómo proceden los usuarios cuando navegan por Internet se realiza la estructura del portal web para que los usuarios accedan a los contenidos de la manera más óptima posible.

5.1.1 Estructura de una página web

Existen miles de formas de disponer de los elementos en un sitio web, pero en estos casos no hay que ser excesivamente creativo en la composición de los elementos ya que tendrán el efecto contrario al buscado y harán que el usuario esté perdido buscando la información que necesita y abandonará el portal web en cuestión de segundos.

Una de las características principales a tener en cuenta es que en la cultura occidental nuestros ojos examinan el contenido de izquierda a derecha y de arriba abajo, por lo que la información principal de una página debe estar concentrada en estar arriba y a la izquierda para que esté presente al usuario a simple vista.

Guiados por estos parámetros de usabilidad los portales web siguen un patrón más o menos fijo para el diseño y estructura de una página.

La imagen 5.1 representa el estructura habitual de una página, en el que destacan las siguientes características:



Figura 5.1 Estructura tipo de una página web.

- **Logotipo:** El logotipo o logo de una web identifica a la empresa u organización a la que pertenece la página web. Debe estar visible en todas las páginas del sitio web. Es la marca con la que te identifica el usuario, por eso debe tener un sitio principal en la estructura de una página web y es por ello que se suele situar en la parte más alta del diseño. También es habitual situar el logotipo dentro de la parte superior a la izquierda, pero eso es algo más flexible dentro del diseño.
- **Navegación:** El sistema de navegación del portal, los menús, debe ser sencillo de encontrar y utilizar para ayudar al visitante del sitio web a moverse sin dificultad dentro de él. Por esta razón también ocupa un lugar importante dentro de la estructura de la página.

Lo más habitual es encontrar el menú de navegación a la izquierda y en la parte central debajo del logotipo de la web. Este diseño ocupa el alrededor del 75% de los sitios web. Es bastante habitual encontrarse diseños donde el menú principal se encuentra a la derecha, es otra opción de diseño útil y que marca una diferencia con la mayoría.

Existe una tercera tendencia de diseño que consiste en tener 3 columnas, donde el menú principal suele estar a la izquierda y a la derecha se sitúa un menú auxiliar. No es lo más habitual pero es una posible solución para

sitios web que tienen mucha navegación.

- **Contenido:** El contenido es la parte principal de una página web, por la que un visitante entra en un portal. Si un usuario no encuentra lo que buscan en cuestión de segundos se irá. El bloque de contenido tiene que ocupar la parte central y más amplia de un diseño.
- **Pie de página:** Se sitúa en la parte inferior de la página. Lo normal es que ofrezca información sobre el contacto y el copyright.
- **Espacios en blanco:** Es una parte muy importante del diseño y que muchos diseñadores pecan de utilizarla, la de tener ausencia de espacios en blanco. Los espacios en blanco ayudan a guiar a los usuarios hacia los bloques donde está el contenido. Una página ocupada al 100% de contenido da la sensación de agobio y no ayuda al usuario a encontrar lo que busca, por lo que cerrará la página a los pocos segundos de entrar y sin encontrar lo que andaba buscando.

Todas estas pautas se crearon bajo la premisa de que los portales web son creados para su consumo en ordenadores personales. Pero con la llegada de los dispositivos móviles y la ampliación de la gama de tamaños de pantalla, el diseño de las aplicaciones web ha tomado una nueva etapa.

5.1.2 Web adaptativa

En el 2010 fue la primera vez que se mencionó *Responsive Web Design*, término que en España se acogió como diseño web adaptativo, que habla de la necesidad de la web actual de adaptarse al dispositivo del que se accede.

En los últimos años, con la aparición de los *smartphones* y *tablets*, la gama de tamaños y resoluciones se ha ampliado por lo que se ha tenido que adaptar el diseño de los sitios web al dispositivo desde que el usuario accede. Esto cambia el concepto de web para los diseñadores, ahora el concepto es el de web única.

Hasta hace unos años los diseñadores concentraban sus esfuerzos en crear una web que se viera idéntica en todos los navegadores y en cualquier pantalla. Como muestra la imagen 5.2, este concepto ha cambiado con la web

adaptativa ya que los diseñadores se centran en crear contenidos que se adapten al medio en el que se van a mostrar, ésto conlleva que una misma web no sea mostrada igual en un monitor de 1024 píxeles de ancho a un móvil de 280 píxeles.

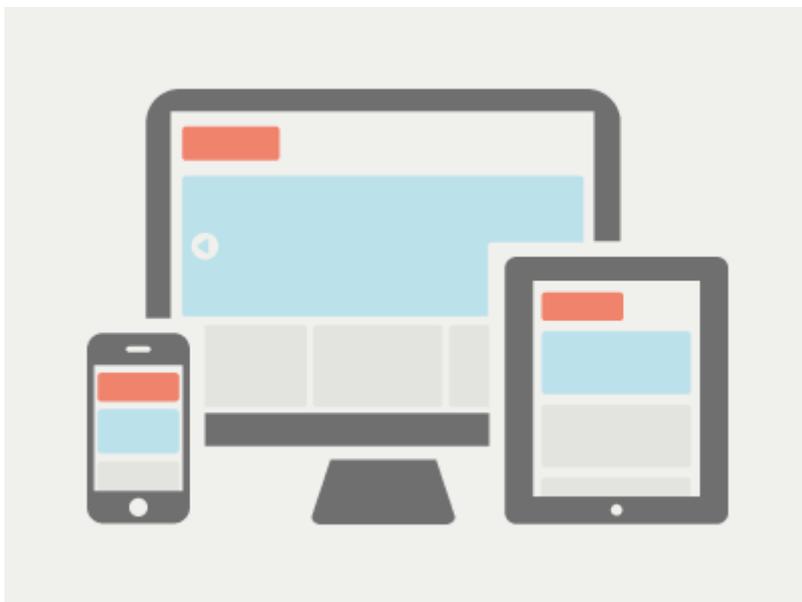


Figura 5.2 Ejemplo de diseño adaptativo.

Todo este esfuerzo de diseño se realiza para que el usuario que accede a un sitio web se encuentre cómodo al ver los contenidos. Es muy poco funcional que un usuario que accede desde un dispositivo móvil vea una web poco legible en la que tiene que desplazarse lateralmente para acceder a todo el contenido que ofrece o tener que estar realizando zoom sobre un texto que quiere leer y volver atrás para seguir viendo el resto de la información.

Para realizar una web adaptativa hay que hacer trabajar sobre la hoja de estilo ya que es el encargado del aspecto del sitio. Se tendrá que decidir a que resolución la web pasa a reducir su número de columnas o que elementos desaparecen a simple vista para mostrarlos en segundo plano. Para realizar estos cambios se necesita trabajar con dimensiones relativas, se deja atrás el trabajar con anchos fijos y se trabaja con porcentajes en relación al ancho de la pantalla.

Otra de las ventajas de realizar una web con diseño responsable recae en el posicionamiento web (SEO). Antes, cuando se creaba una versión web para móviles el tráfico se dividía entre los que accedían a la web desde un ordenador o portátil y los que lo hacía desde los móviles o tablets, ésto hacía que el tráfico se

dividiera entre las dos versiones de una misma web. Ahora al tener una única URL el tráfico es el mismo por lo que los resultados son mayores y la página aparecerá más arriba en los resultados de los buscadores.

Hoy en día no se concibe crear una nueva web sin plantearse hacer una web de diseño responsable. Los más importantes sistemas de gestión de contenidos como son WordPress, Joomla o Drupal ya ofrecen un gran número de plantillas de diseños adaptativos. Con todo ello se estima que sólo un 10% de las webs actuales tienen un diseño adaptado a todos los tipos de dispositivos.[6]

5.1.3 Diseño inicial del portal web

Teniendo en mente que reglas son las que hay que cumplir para el diseño del portal web se ha utilizado el programa Balsamiq Mockups para realizar un *wireframe*. Un *wireframe* en diseño web se conoce como *Storyboard* o plano de pantalla.

Los *wireframes* sirven como herramienta de comunicación y discusión entre programadores, diseñadores y clientes. La gran ventaja de utilizar *wireframes* es que se define la organización y estructura de la web antes de la programación y con ello se evita cambios posteriores más costosos consiguiendo la reducción de los tiempos de creación.

- **Diseño de la web con balsamiq**

Con Balsamiq Mockups se procede a crear el *wireframe* del portal web. Se realizan los bocetos sin muchos detalles de la organización y el aspecto general que debe tener la web tomando en cuenta todo lo dicho con anterioridad.

- Estructura general del portal web.
- Ubicación de la navegación de la web. Menú principal y el menú de usuarios.
- Contenido que se incluye en la web.
- La distribución del contenido para que la información sea clara y directa al

usuario.

A continuación se muestran los bocetos creados con Balsamiq que inician el proceso de generación del portal web.

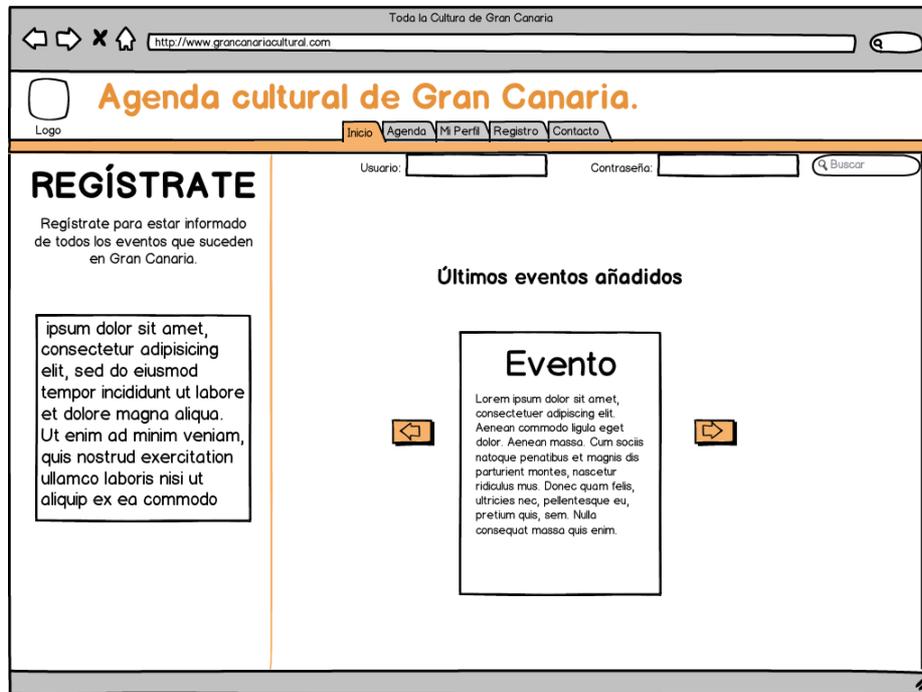


Figura 5.3 Boceto de la página inicio con Balsamiq

En la imagen 5.3 anterior está definida la página de inicio del portal web. Esta página se puede dividir en dos partes:

- Por un lado está la parte común del portal web que son el logotipo, el título de la web, menú principal y la entrada de usuarios registrados. Están situados en la parte superior de la web y son visibles en todas las páginas del portal web.
- La zona única perteneciente a la página de inicio está debajo del menú principal. Aquí se encuentra una parte que da acceso directo a los visitantes a registrarse en el sistema y la otra que informa de los últimos eventos creados en la web.

Las dos siguientes imágenes, figuras 5.4 y 5.5, están relacionadas con la sección agenda dentro de la web, en esta sección se muestran los eventos de la agenda cultural.

En la primera imagen se muestra un calendario para buscar los eventos según la fecha y un buscador para que un usuario busque un evento en concreto. Además existen dos menús, uno a la izquierda en el que se accede a los eventos según al municipio al que pertenezca y otro debajo del menú principal para acceder según el tipo de evento que corresponda.



Figura 5.4 Boceto de la página agenda con Balsamiq

Accediendo al menú de ayuntamientos se ven todos los eventos que pertenecen a un ayuntamiento en concreto. Se muestra información correspondiente del evento, imagen del evento, si la hubiese y comentarios de los usuarios.



Figura 5.5 Boceto de la página agenda-2 con Balsamiq

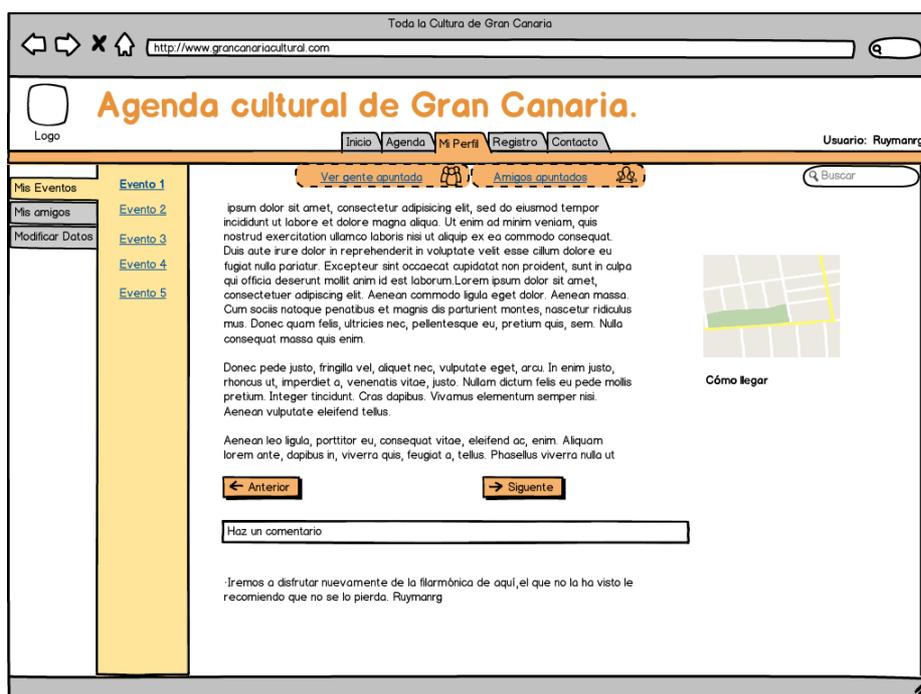


Figura 5.6 Boceto de la página mi perfil con Balsamiq

El menú de usuario de la imagen 5.6 muestra información personal de un usuario registrado en la web. Informa de los eventos a los que se ha apuntado y los datos personales dando la posibilidad de cambiarlos en cualquier momento.



Figura 5.7 Boceto de la página registro con Balsamiq

Las dos últimas imágenes, figuras 5.7 y 5.8, corresponden a las secciones Registro y Contacto donde el usuario se comunica con el sistema a través de los formularios de entrada. En registro un usuario rellena los campos con sus datos personales para darse de alta en el sistema mientras que en contacto cualquier visitante de la web informa a la administración de la web ante cualquier incidencia.

Estos son los bocetos realizados con Balsamiq para hacer una idea general de cómo será la web. El resultado final del portal web será un fiel reflejo de lo expuesto en estos bocetos, aunque habrá cambios más relacionados con el aspecto de la web que con la funcionalidad de la misma.

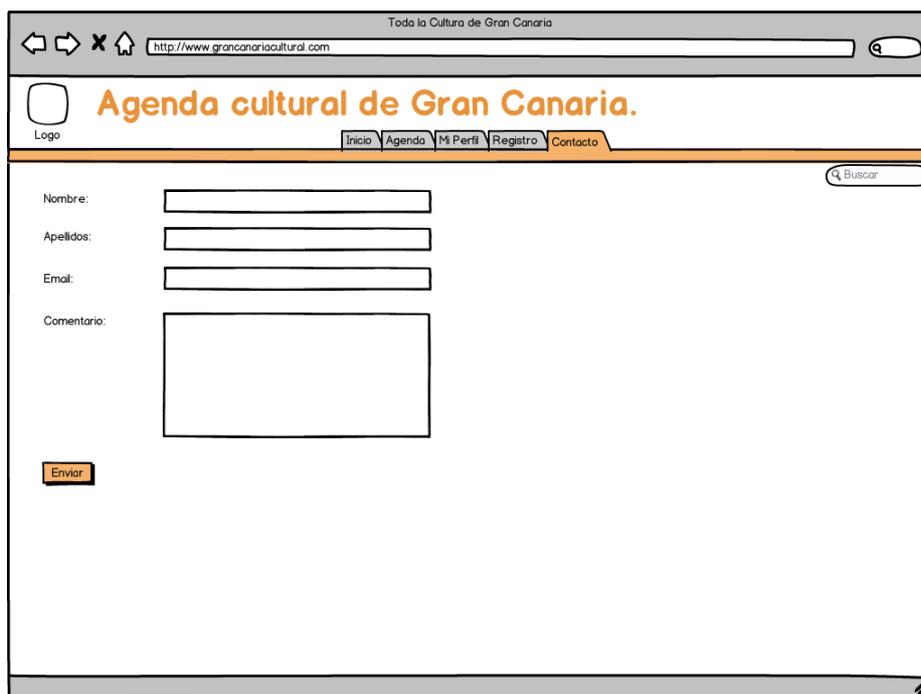


Figura 5.8 Boceto de la página contacto con Balsamiq

5.1.4 Estructura de la web

Una página cultural como la que se va a crear tiene como usuarios potenciales a un amplio espectro de la sociedad, desde personas cuya cultura tecnológica es avanzada hasta personas que no sean demasiado hábiles en el mundo de navegar por sitios web. Debido a este amplio espectro de usuarios se ha buscado que tanto el diseño y sobre todo la estructura del portal web sea sencilla.

La web cuenta con un menú accesible desde cualquier punto dentro del portal para que el usuario pueda cambiar de contenido de una manera rápida y sencilla. Este menú tiene 6 apartados bien diferenciados:

- **Inicio.** Contiene la página principal del portal web. En ella se mostrarán los últimos diez eventos creados en la web.
- **Agenda.** Se muestran los veintitún municipios de Gran Canaria y accediendo a cada uno de ellos se puede ver todos los eventos que tienen programados.

- **Calendario.** Incluye una vista de un calendario resaltando los días en los que hay eventos programados.
- **Usuario.** Sección de los usuarios registrados para poder acceder al menú de usuario.
- **Registrarse.** Muestra un formulario para que los usuarios se registren en el portal web.
- **Contacto.** Incluye otro formulario para que cualquier visitante de la web informe a los administradores del portal de cualquier incidencia.

Además en todas las secciones principales se incluye una barra lateral accesible desde cualquier página del portal web que indica los eventos más cercanos en el tiempo a realizarse indistintamente del municipio donde vaya a tener lugar.

La visualización de los eventos se ha querido realizar también de forma lo más clara posible al usuario para fomentar su total comprensión. La información que componen los eventos es:

- El título del evento, la fecha y hora de cuando se va a producir.
- Texto explicativo que incluirá un mapa que mostrará el punto exacto donde se realizará el evento.
- Una imagen que acompañe al texto que habitualmente mostrará el cartel del evento.

En la figura 5.9 se muestra como es la visualización típica de un evento en la web. En el capítulo 3 del Pliego de condiciones se presentará el manual de usuario de las distintas secciones del portal web.



Inicio Agenda Calendario Usuario Contacto Registrarse

XI Feria del Sureste. 7, 8 y 9 de noviembre de 2014

Subido el día November 2, 2014 by roymang

Este evento será el día **7 de noviembre de 2014**
Y comenzará a las **10:00**

Los municipios de Agüimes, Ingenio y Santa Lucía, conscientes de que la mayoría de los turistas desconocen la riqueza cultural, tradicional y el propio entorno natural de la Isla que no corresponde a las zonas turísticas, decidieron desde el año 2004 realizar la Feria del Sureste, donde los participantes puedan conocer y disfrutar de los atractivos que ofrecen estos lugares a través de las tradiciones y costumbres de sus habitantes, la artesanía, la agricultura, la ganadería, el folklore y la alimentación (el cochino negro canario, la elaboración de aceite de oliva, la moda calada, los quesos y mojos que se exportan fuera de nuestras islas, los tan premiados vinos, la apiterapia, etc., seguramente por separado o en su conjunto sorprenden al visitante foráneo y local).

La Feria, promueve la actividad económica del turismo de los lugares cuya orografía y comunicaciones no les brindan muchas oportunidades. Se ha convertido, tras 7 años de realización, en una magnífica oportunidad para Artesanos, Agricultores, Ganaderos y Comercios del Turismo Rural que muestran y venden sus productos en un bello ambiente favorecedor donde se mezclan sabores y ritmos musicales, y que se ubica estratégicamente cada año, de manera alterna, en lugares de fácil acceso y que son una de las principales vías comerciales de cada municipio: Avda. de Ansite, en el Cruce de Arinaga, T.M. Agüimes; calle Carlos V, en Carrizal, T.M. de Ingenio; y Avda. de Canarias, en Vecindario, T.M. de Santa Lucía.

Este año se realizará los días 7, 8 y 9 de noviembre en la avenida de Ansite, Cruce de Arinaga, en el municipio de Agüimes. Es posible gracias al entusiasmo, apoyo y participación de los agentes económicos y ciudadanía en general, así como por la valiosa colaboración de la Consejería de Empleo, Industria, Comercio y Artesanía del Cabildo de Gran Canaria.

Cada año tiene una duración de un fin de semana, para facilitar la participación de los expositores. En todas las ferias celebradas se ha pretendido una perfecta imbricación de la misma con las actividades comerciales de la zona, logrando que lejos de representar un problema sea un elemento de beneficio mutuo.

Este evento está postado en Eventos de Agüimes.

← La hora del cuento. Biblioteca Insular

Figura 5.9 Ejemplo de un evento

5.1.5 Plantilla de WordPress

Definidos los patrones de diseño y de estructura y teniendo una primera idea sobre cómo se va a organizar el portal web, se procede a buscar una plantilla de WordPress que cumpla con las necesidades del proyecto.

Este proceso de selección de la plantilla tuvo un paso previo que fue el de la elección de WordPress como el sistema de gestor de contenidos con el que se trabajará para la creación del sitio web. Las plantillas de WordPress son las encargadas de darle el diseño a una aplicación web.

La plantilla seleccionada tiene que cumplir varios requisitos principales:

- Ser una plantilla con diseño adaptativo, condición primordial para la elección ya que se quiere que el proyecto esté adaptado a los últimos avances y sea perfectamente visible en toda la gama de dispositivos del mercado.
- Cumpla con los parámetros principales de usabilidad de un sitio web para que los usuarios que accedan a la web desde los ordenadores.
- La plantilla seleccionada tiene que ser gratuita porque es una de las premisas iniciales de nuestro proyecto.

La búsqueda de una plantilla ideal para WordPress puede ser una tarea que ocupe un gran espacio de tiempo debido a que la oferta es realmente amplia, en el momento de escribir estas líneas, en la web oficial de WordPress ofertan 2.530 plantillas gratuitas diferentes.

Una primera selección redujo la elección de plantillas a tres. *Gridster-Lite*, *Catch Box* y *Coller*, que son mostradas en la figura 5.10, dónde se exponen las 3 plantillas seleccionadas inicialmente. Todas tenían en común ser de diseño responsable y gratuitas. En principio ya entraba más en el tema estético la elección de una sobre otra.

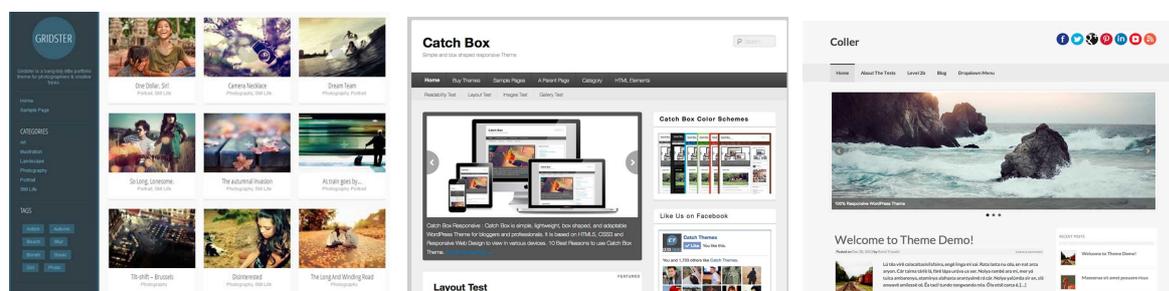


Figura 5.10 Imagen de las tres plantillas seleccionadas

Después de instalarlas en el servidor local de WordPress y trabajar con ellas para ver las posibilidades que nos ofrecían se decidió descartar la plantilla Gridster-Lite ya que no era la plantilla ideal para lo que se necesitaba. Era una plantilla más especializada en crear portafolio de fotos y no ofrecía la posibilidad de incluir una segunda barra lateral.

Finalmente la plantilla elegida fue la plantilla Coller porque respondía a todas necesidades estructurales como de diseño. Como se comentó era una plantilla de diseño flexible, gratuita, en la que la información se mostraba de forma clara, teniendo la posibilidad de crear barra de navegación laterales a ambos lados y un menú de navegación central apropiado para las secciones pensadas en nuestro boceto inicial.

A partir de la elección de la plantilla ya comienza la tarea de ir añadiendo funcionalidades adicionales a través de los *plugins* propios de WordPress y del desarrollo de código propio.

5.1.6 Creación del logotipo

Otra de las partes dedicadas al diseño consiste en la creación de un logotipo o logo. El logotipo es utilizado por las empresas, sean o no portales web, para identificarse ante los usuarios y diferenciarse de la competencia.

La creación del logotipo en el proyecto tiene una doble función:

- En la parte web realiza la función de identificación en el portal web
- En la parte de la aplicación realiza la misma función de identificación en el menú de la aplicación pero además se utiliza el logotipo como icono de aplicaciones, que es como se conoce a los iconos de acceso para abrir las aplicaciones en los móviles.

El logotipo diseñado, representado en la imagen 5.11 de la siguiente página, está basado en la silueta de la isla de Gran Canaria en las que sobresalen las letras “GCC” que son las iniciales de “Gran Canaria cultural” que es el nombre

de portal web y de la aplicación móvil. Estas letras están coloreadas con los colores azul y amarillo de la bandera oficial de la isla de Gran Canaria.



Figura 5.11 Logotipo del proyecto.

5.2 Comprobación de la web en varios navegadores

Uno de los grandes problemas a los que se enfrenta un programador a la hora de hacer de portales web es la compatibilidad con todos los navegadores del mercado, o al menos con los más usados por los usuarios.

Hacer que una web sea compatible ayuda a una buena experiencia de los usuarios en la web, para hacer que la web sea compatible al mayor número de usuarios posibles se debe realizar un estudio de mercado para ver que navegadores son los más utilizados.

5.2.1 Estudio del uso de navegadores

Se realizó un estudio para comprobar cuales eran los navegadores más utilizados del mercado para hacer una comprobación y ver que el portal web era compatible con ellos. Hay que tener en cuenta que además de gran cantidad de navegadores existen versiones diferentes de cada uno de ellos.

Se considera una buena práctica que un programador pruebe el diseño de su web en aquellos navegadores que copen aproximadamente el 90% de la cuota de mercado. A primera vista se puede deducir que los más utilizados son Internet Explorer, Google Chrome y Mozilla Firefox pero se buscarán datos reales para confirmar esta suposición.

Buscar información sobre qué estudio es el más fidedigno sobre el uso de navegadores es casi imposible ya que no hay un estudio oficial pero si existen webs con prestigio sobre la que buscar los resultados[7][8]. Éstas son las webs de referencia que se han utilizado para obtener los resultados debido a que según sus informaciones entre la suma de los dos estudios analizan los resultados de más de 3,5 millones de usuarios en el mundo.

Los estudios muestran unos resultados radicalmente distintos entre un estudio y otro. Para el portal StatCounter el navegador más utilizado mayoritariamente es Google Chrome como se ve en la figura 5.12, mientras que en la imagen 5.13 se muestra que para el portal NetMarketShare Internet Explorer es el favorito para los usuarios.

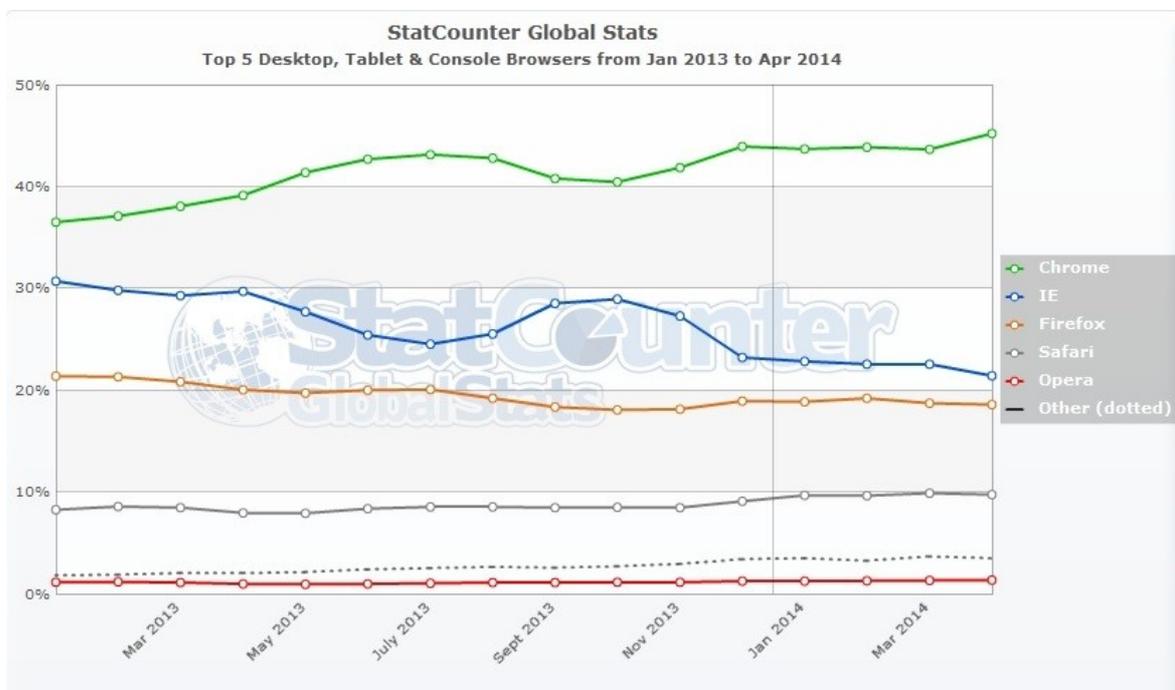


Figura 5.12 Gráfica de cuota de mercado de navegadores

Estos datos son a nivel mundial sin tener en cuenta diferentes sistemas operativos ni diferentes versiones dentro del mismo navegador. Ambos estudios se refieren al periodos comprendido entre Enero de 2013 y Abril de 2014.

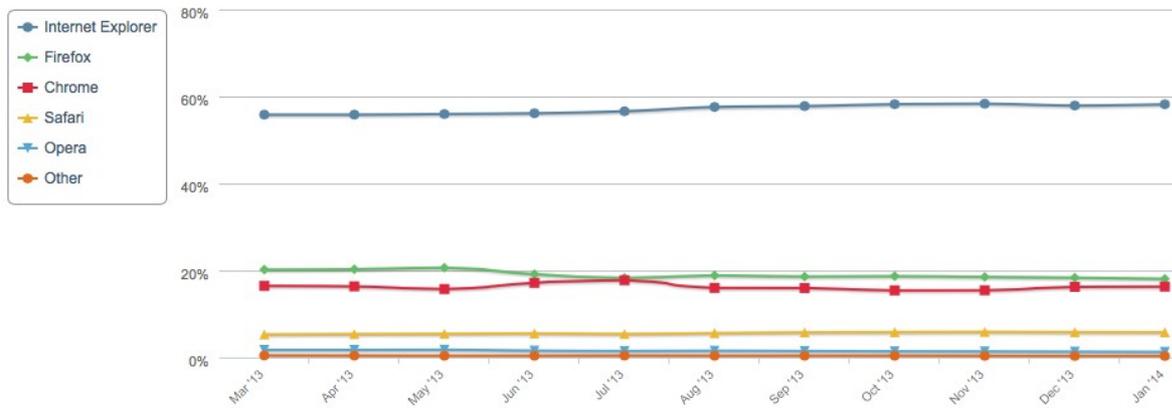


Figura 5.13 Gráfica de cuota de mercado de navegadores

Respecto a los navegadores utilizados en España sólo se han obtenido los resultados de la web StatCounter que muestran un uso mayor de Google Chrome entre los usuarios españoles como se puede comprobar en la figura 5.14 mostrada a continuación.

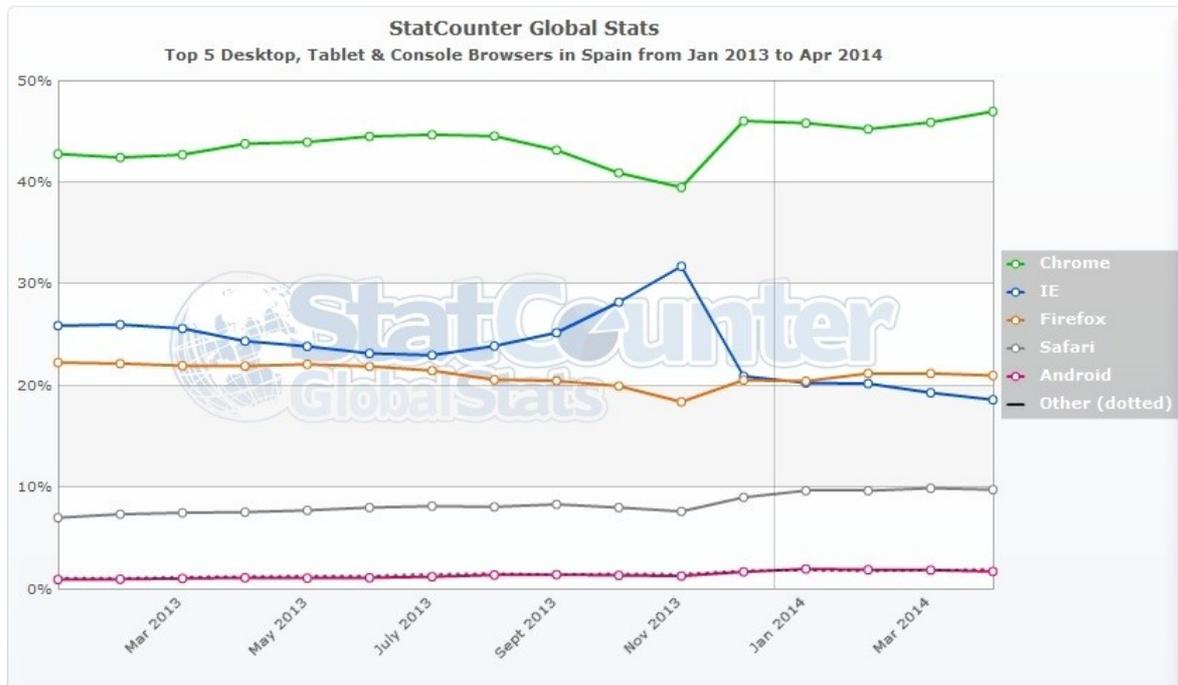


Figura 5.14 Gráfica de cuota de mercado de navegadores en España

Los datos que nos muestran estos estudios aunque muy distintos a nivel de liderazgo individual de un navegador sobre otro son de verdadera ayuda para el estudio que se está realizando ya que nuestra búsqueda, como se comentó con anterioridad, es buscar la suma de los navegadores que ocupan el 90% de la cuota de mercado y según los datos que ofrecen podemos sacar las siguientes conclusiones:

- La suma de los navegadores Internet Explorer, Google Chrome y Mozilla Firefox son los que utilizan los usuarios en más de un 90% de los casos.
- Es importante hacer la comprobación del portal web en el navegador Safari ya que es el navegador utilizado por los usuarios de Apple y que ocupa en España cerca de 10% de los usuarios a los que no hay que olvidar.

5.2.2 Comprobación de compatibilidad del portal web

Extraídas las conclusiones del uso de los navegadores se procede a la instalación de los navegadores Google Chrome, Mozilla Firefox y Safari en su versión para Windows, además del Internet Explorer que viene por defecto en el sistema operativo Windows.

Cabe destacar que para la comprobación de los distintos navegadores existen aplicaciones que simulan como se vería un portal web en múltiples navegadores y en distintas versiones de los mismos pero no se eligió esta opción ya que una de las recomendaciones al afrontar un proyecto de un portal web es limitar la cantidad de navegadores y versiones a los que está optimizado tu portal web, ya que es prácticamente imposible estar optimizados para el 100% de navegadores y versiones que existen en el mercado.

Las pruebas de compatibilidad del portal web se realizaron con los siguiente navegadores utilizándose la última versión disponible para cada uno de ellos:

- Internet Explorer, versión 11.0.9600.17105.
- Google Chrome, versión 34.0.1847.131 m.

- Mozilla Firefox, versión 29.0.1.
- Safari, versión 5.1.7.
- Opera, versión 25.0.1614.68.

En las siguientes figuras se ven diferentes páginas del portal web donde se puede observar que la visualización es igual indistintamente del navegador usado por el usuario. Estas imágenes están tomadas bajo el sistema operativo Windows y con los navegadores Opera y Chrome mencionados anteriormente.



Figura 5.15 Página Inicio para Opera y Chrome

En la imagen 5.15 se ve como representa cada navegador la página de inicio del portal web, y se muestra que efectivamente no hay diferencia entre un navegador y otro. Igual pasa en la imagen 5.16, dónde se representa la sección Agenda del portal web.

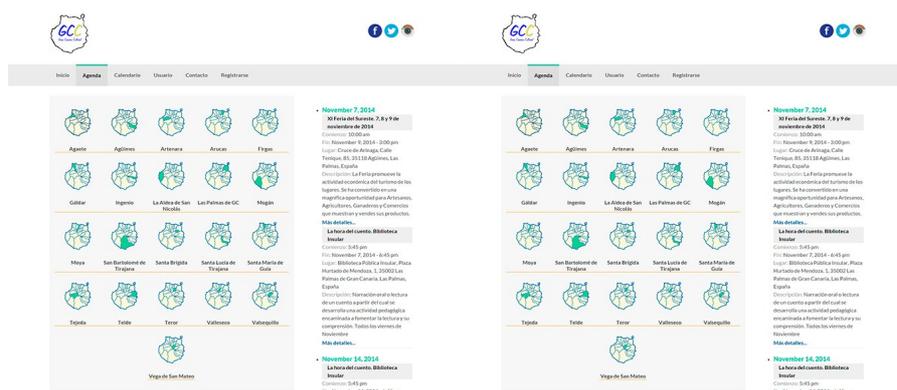


Figura 5.16 Página Agenda para Opera y Chrome

La imagen 5.17 muestra la sección Calendario del portal web en ambos navegadores. Ambas imágenes son idénticas por lo que se comprueba que el diseño del portal web es compatible con los navegadores más utilizados del mercado.

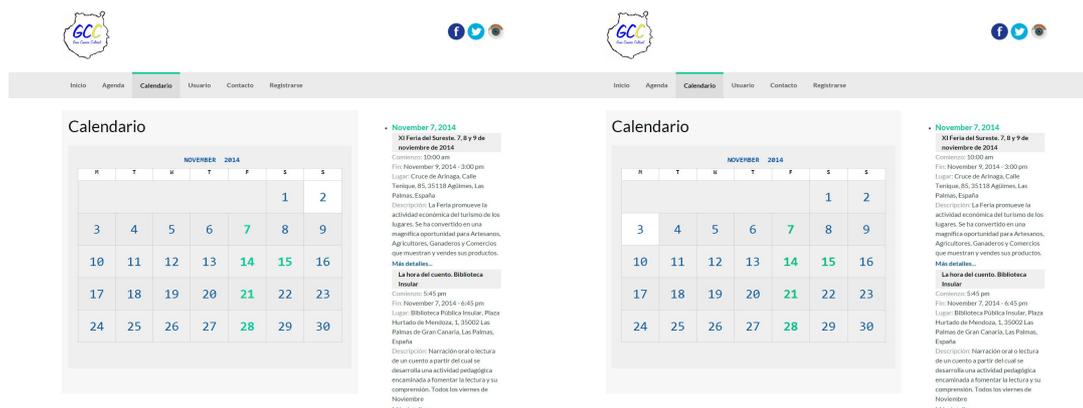


Figura 5.17 Página Calendario para Opera y Chrome

5.2.3 Navegadores para dispositivos móviles

Al hacer hincapié en crear una aplicación web adaptativa hay que realizar el estudio de los navegadores web para móviles y ver si nuestra web se adapta convenientemente a otras resoluciones y tamaños de pantalla.

Se vuelve a hacer un estudio mirando los navegadores más utilizados para móviles y según los resultados comprobar nuestro portal web en aquellos navegadores que sean los que copan el mercado. Se toma las mismas referencias que para los navegadores de escritorio. Y nos arrojan los siguientes

resultados representados en las siguientes imágenes.

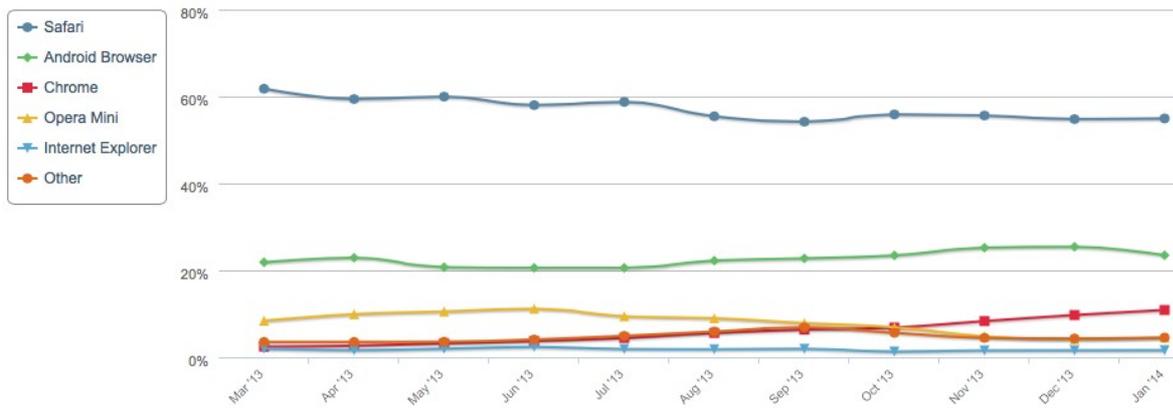


Figura 5.18 Gráfica de cuota de mercado de navegadores móviles

Tanto en la imagen 5.18 como en la 5.19 se puede comprobar que la suma de los navegadores Safari (principalmente iPhone), Chrome y el navegador de Android que trae por defecto este sistema operativo ocupan la gran mayoría del mercado en el último año. Con estos datos en la mano se realizará el estudio de estos tres navegadores para comprobar que realmente el diseño de nuestro sitio web es adaptativo.

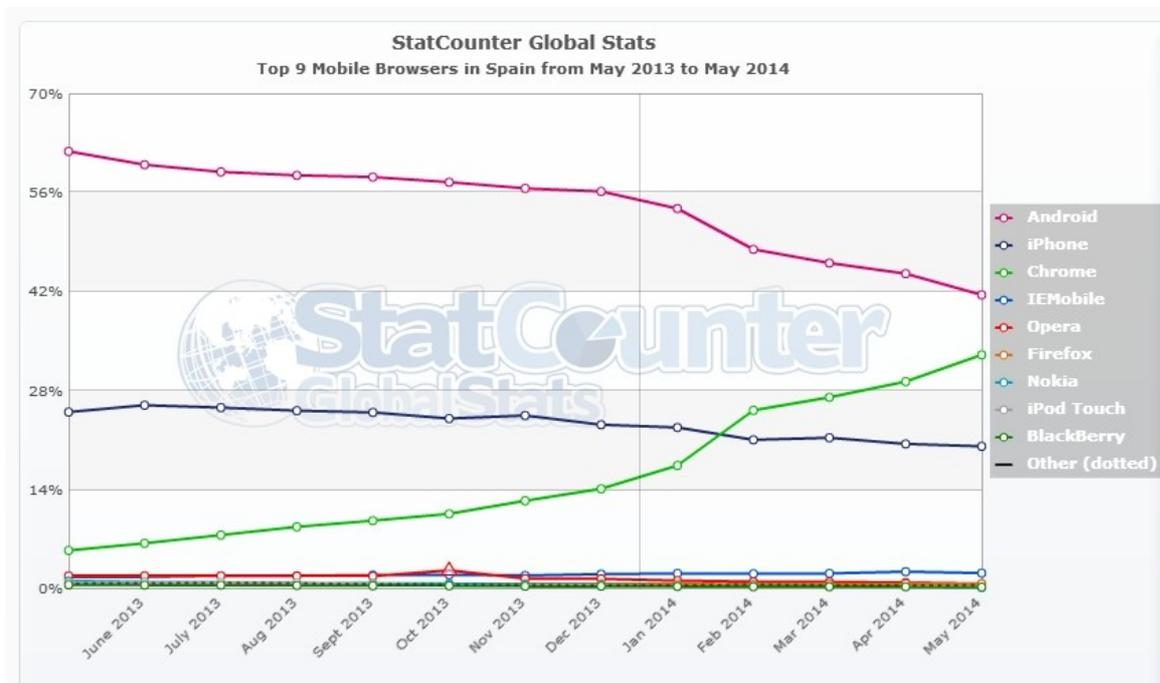


Figura 5.19 Gráfica de cuota de mercado de navegadores móviles

Las pruebas de compatibilidad de nuestro portal se han hecho con los

siguiente navegadores en combinación con el modelo de móvil y el sistema operativo instalado en éstos en el momento de la comprobación.

- Navegador de Android en un móvil Samsung GALAXY S III con el S.O. Android 4.3.
- Navegador Chrome, versión 35.0.1916.138 en un móvil Samsung GALAXY S III con el S.O. Android 4.3.
- Navegador Safari en un móvil iPhone 4 con el S.O. ios 7.1.1.
- Navegador Opera, versión 7.0.5.45389 en un móvil iPhone 4 con el S.O. ios 7.1.1.

Para completar la comprobación de los navegadores móviles se hizo en el navegador de un tablet. Y el resultado fue positivo en el siguiente dispositivo.

- Navegador de Android en un tablet Woxter Tablet PC 90 BL con el S.O. Android 4.0.4.

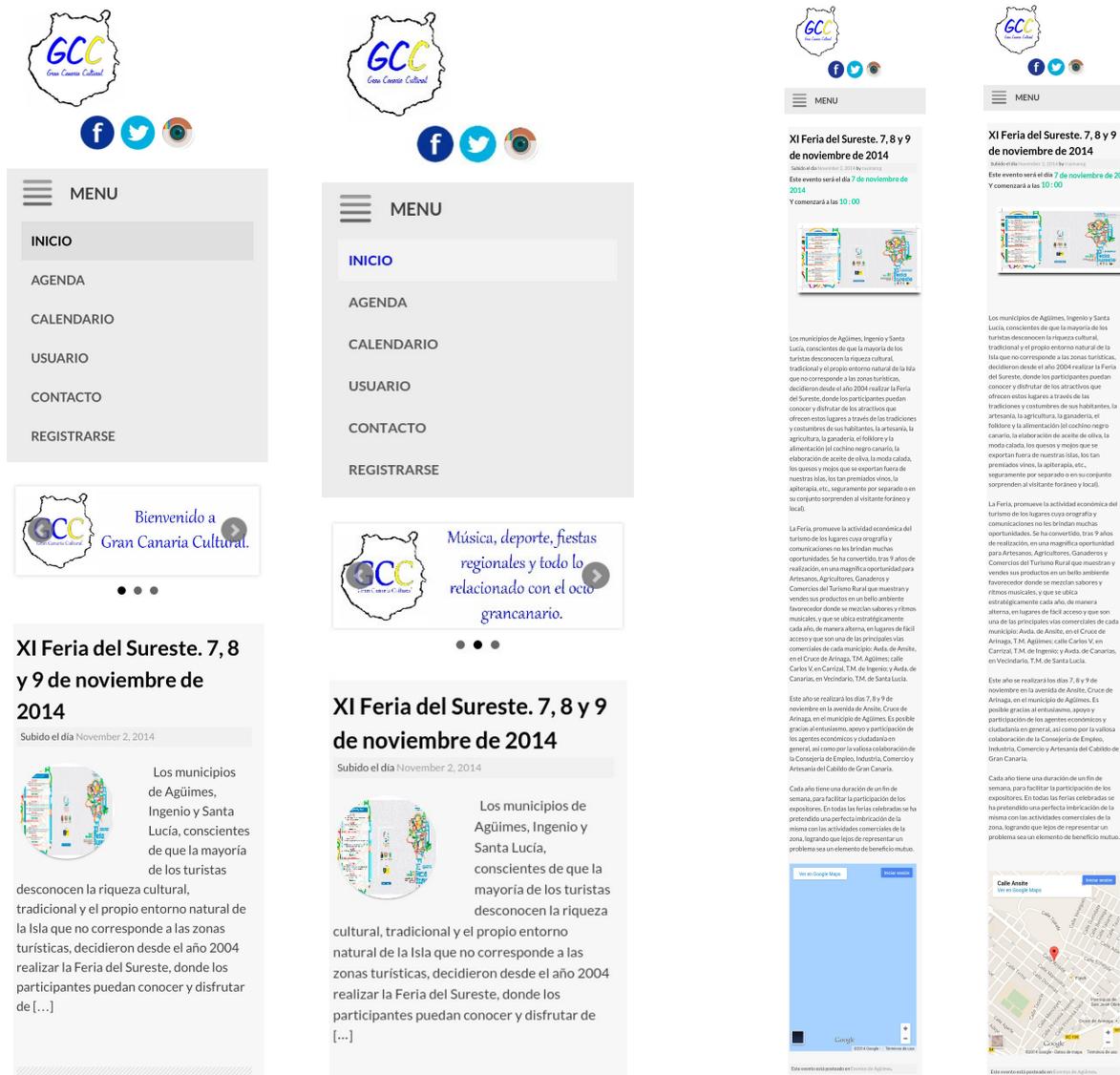


Figura 5.20 Página Inicio y evento para iPhone4 y SamsungS3

En la figura anterior 5.20 se muestra como se ven diferentes partes de la web en versión para móviles, en este caso un iPhone 4 y un Samsung Galaxy S3. La imagen izquierda muestra la página de inicio con el menú desplegado y la imagen de la derecha muestra el formato de cómo se ve un evento en los 2 móviles. Igualmente en la imagen 5.21 posterior se ve la página que muestra el calendario y el *widjets* con los próximos eventos y se comprueba que la representación en ambos móviles es igual.



GCC
Gran Canaria Cultural



GCC
Gran Canaria Cultural









☰ MENU

☰ MENU

Calendario

NOVEMBER 2014

M	T	W	T	F	S	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

Calendario

NOVEMBER 2014

M	T	W	T	F	S	S
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

November 7, 2014
XI Feria del Sureste, 7, 8 y 9 de noviembre de 2014
Comienzo: 10:00 am
Fin: November 9, 2014 - 3:00 pm
Lugar: Cruce de Arriaga, Calle Tenorio, 85, 35118 Agüimes, Las Palmas, España
Descripción: La Feria promueve la actividad económica del turismo de los lugares. Se ha convertido en una magnífica oportunidad para Artesanos, Agricultores, Ganaderos y Comercios que muestran y venden sus productos.
Más detalles...

La hora del cuento. Biblioteca Insular
Comienzo: 5:45 pm
Fin: November 7, 2014 - 6:45 pm
Lugar: Biblioteca Pública Insular, Plaza Hurtado de Mendoza, 1, 35002 Las Palmas de Gran Canaria, Las Palmas, España
Descripción: Narración oral o lectura de un cuento a partir del cual se desarrolla una actividad pedagógica encaminada a fomentar la lectura y su comprensión. Todos los viernes de Noviembre.
Más detalles...

November 14, 2014
La hora del cuento. Biblioteca Insular
Comienzo: 5:45 pm
Fin: November 14, 2014 - 6:45 pm
Lugar: Biblioteca Pública Insular, Plaza Hurtado de Mendoza, 1, 35002 Las Palmas de Gran Canaria, Las Palmas, España
Descripción: Narración oral o lectura de un cuento a partir del cual se desarrolla una actividad pedagógica encaminada a fomentar la lectura y su comprensión. Todos los viernes de Noviembre.
Más detalles...

November 15, 2014
Pig Feet
Comienzo: 8:30 pm
Fin: November 16, 2014 - 2:00 am
Lugar: Carrizal, Las Palmas, España
Descripción: El sábado 15 de noviembre en la Plaza del Buen Suceso (Carrizal - Ingenio), a partir de las 20:00 horas con: Del con Dios y Alameda de Soubilla y otras tres bandas canarias: Mondo Diavolo (GC), Dactyl Chundo (TH) y Mumbator (PV).
Más detalles...

November 21, 2014
La hora del cuento. Biblioteca Insular
Comienzo: 5:45 pm
Fin: November 21, 2014 - 6:45 pm
Lugar: Biblioteca Pública Insular, Plaza Hurtado de Mendoza, 1, 35002 Las Palmas de Gran Canaria, Las Palmas, España
Descripción: Narración oral o lectura de un cuento a partir del cual se desarrolla una actividad pedagógica encaminada a fomentar la lectura y su comprensión. Todos los viernes de Noviembre.
Más detalles...

November 21, 2014
La hora del cuento. Biblioteca Insular
Comienzo: 5:45 pm
Fin: November 21, 2014 - 6:45 pm
Lugar: Biblioteca Pública Insular, Plaza Hurtado de Mendoza, 1, 35002 Las Palmas de Gran Canaria, Las Palmas, España
Descripción: Narración oral o lectura de un cuento a partir del cual se desarrolla una actividad pedagógica encaminada a fomentar la lectura y su comprensión. Todos los viernes de Noviembre.
Más detalles...

November 28, 2014
La hora del cuento. Biblioteca Insular
Comienzo: 5:45 pm
Fin: November 28, 2014 - 6:45 pm
Lugar: Biblioteca Pública Insular, Plaza Hurtado de Mendoza, 1, 35002 Las Palmas de Gran Canaria, Las Palmas, España
Descripción: Narración oral o lectura de un cuento a partir del cual se desarrolla una actividad pedagógica encaminada a fomentar la lectura y su comprensión. Todos los viernes de Noviembre.
Más detalles...

November 7, 2014
XI Feria del Sureste, 7, 8 y 9 de noviembre de 2014
Comienzo: 10:00 am
Fin: November 9, 2014 - 3:00 pm
Lugar: Cruce de Arriaga, Calle Tenorio, 85, 35118 Agüimes, Las Palmas, España
Descripción: La Feria promueve la actividad económica del turismo de los lugares. Se ha convertido en una magnífica oportunidad para Artesanos, Agricultores, Ganaderos y Comercios que muestran y venden sus productos.
Más detalles...

La hora del cuento. Biblioteca Insular
Comienzo: 5:45 pm
Fin: November 7, 2014 - 6:45 pm
Lugar: Biblioteca Pública Insular, Plaza Hurtado de Mendoza, 1, 35002 Las Palmas de Gran Canaria, Las Palmas, España
Descripción: Narración oral o lectura de un cuento a partir del cual se desarrolla una actividad pedagógica encaminada a fomentar la lectura y su comprensión. Todos los viernes de Noviembre.
Más detalles...

November 14, 2014
La hora del cuento. Biblioteca Insular
Comienzo: 5:45 pm
Fin: November 14, 2014 - 6:45 pm
Lugar: Biblioteca Pública Insular, Plaza Hurtado de Mendoza, 1, 35002 Las Palmas de Gran Canaria, Las Palmas, España
Descripción: Narración oral o lectura de un cuento a partir del cual se desarrolla una actividad pedagógica encaminada a fomentar la lectura y su comprensión. Todos los viernes de Noviembre.
Más detalles...

November 15, 2014
Pig Feet
Comienzo: 8:30 pm
Fin: November 16, 2014 - 2:00 am
Lugar: Carrizal, Las Palmas, España
Descripción: El sábado 15 de noviembre en la Plaza del Buen Suceso (Carrizal - Ingenio), a partir de las 20:00 horas con: Del con Dios y Alameda de Soubilla y otras tres bandas canarias: Mondo Diavolo (GC), Dactyl Chundo (TH) y Mumbator (PV).
Más detalles...

November 21, 2014
La hora del cuento. Biblioteca Insular
Comienzo: 5:45 pm
Fin: November 21, 2014 - 6:45 pm
Lugar: Biblioteca Pública Insular, Plaza Hurtado de Mendoza, 1, 35002 Las Palmas de Gran Canaria, Las Palmas, España
Descripción: Narración oral o lectura de un cuento a partir del cual se desarrolla una actividad pedagógica encaminada a fomentar la lectura y su comprensión. Todos los viernes de Noviembre.
Más detalles...

November 21, 2014
La hora del cuento. Biblioteca Insular
Comienzo: 5:45 pm
Fin: November 21, 2014 - 6:45 pm
Lugar: Biblioteca Pública Insular, Plaza Hurtado de Mendoza, 1, 35002 Las Palmas de Gran Canaria, Las Palmas, España
Descripción: Narración oral o lectura de un cuento a partir del cual se desarrolla una actividad pedagógica encaminada a fomentar la lectura y su comprensión. Todos los viernes de Noviembre.
Más detalles...

November 28, 2014
La hora del cuento. Biblioteca Insular
Comienzo: 5:45 pm
Fin: November 28, 2014 - 6:45 pm
Lugar: Biblioteca Pública Insular, Plaza Hurtado de Mendoza, 1, 35002 Las Palmas de Gran Canaria, Las Palmas, España
Descripción: Narración oral o lectura de un cuento a partir del cual se desarrolla una actividad pedagógica encaminada a fomentar la lectura y su comprensión. Todos los viernes de Noviembre.
Más detalles...

Figura 5.21 Página del Calendario y widget de eventos para iPhone4 y SamsungS3

Diseño y estructura de la aplicación

Para la creación de la aplicación móvil se ha utilizado un *framework* creado por el Instituto Universitario de Microelectrónica Aplicada (IUMA) de la Universidad de Las Palmas de Gran Canaria.

Este *framework* ha sido creado para ayudar a la creación de aplicaciones móviles multisistema. Como todo *framework* es un software que ha sido diseñado para ayudar a la creación de otros software, en este caso, aplicaciones móviles.

El sistema está optimizado para trabajar con el plugin de WordPress *SpiderCatalog*. El inconveniente con este plugin es que trabaja sobre una base de datos en MySQL que modifica para crear su sistema de archivos y Heroku despliega las aplicaciones en el sistema PostgreSQL por lo que es incompatible utilizar este plugin. Para utilizar el sistema creado por IUMA se ha realizado la tarea de adaptar código en PHP y la creación de tablas en Parse para simular el funcionamiento de este plugin. Este capítulo está dividido en tres 3 apartados:

- Se realiza una introducción al plugin *SpiderCatalog* para explicar los conceptos con los que se va a trabajar.
- Desarrollo del sistema creado para el sistema partiendo de la estructura utilizada por *SpiderCatalog*.
- Se desarrolla el trabajo realizado para crear un ambiente similar al plugin para la utilización del *framework* creado por el IUMA.
- Se muestra el resultado final de la aplicación.

6.1 SpiderCatalog

Spider Catalog es un plugin de WordPress que facilita la organización jerarquizada de productos por catálogos. Para realizar esta organización utiliza dos elementos principales que son las categorías y los productos. Cada producto pertenece a una o varias categorías en las que se divide el catálogo.

Las categorías que componen la aplicación está constituida por diferentes

niveles jerárquicos que ayudan a la organización. Las categorías tienen diferentes parámetros donde podemos ver el nombre de la categoría, la descripción, las subcategorías que tiene, la categoría a la que pertenece, el numero de orden si es pública o no. En la tabla 6.1 se muestran los parámetros de categorías.

Categorías	
Name	Nombre de la categoría
Description	Descripción de la categoría
Image_url	Url de la imagen que se asocia a la categoría
Parent	Categoría a la que pertenece
Param	Parámetros que definen a una categoría
Id	Identifica a una categoría
Ordering	Orden de una categoría
Published	Categoría publicada

Tabla 6.1 Parámetros de categorías de SpiderCatalog

Los productos son los elementos del catálogo que son clasificados dependiendo a la categoría o categorías que pertenezcan. Al igual que las categorías los productos tienen unos parámetros que los definen como son el nombre del producto, la descripción, el precio, las categorías a las que pertenece, el numero de Orden o si es pública o no. En la tabla 6.2 se muestran los parámetros de productos.

Productos	
Name	Nombre del producto
Description	Descripción del producto
Image_url	Url de la imagen que se asocia al producto
Parent	Categoría a la que pertenece el producto
Param	Parámetros que definen a un producto
Category_Id	Categorías a la que pertenece un producto
Ordering	Orden de un producto
Published	Producto publicado

Tabla 6.2 Parámetros de productos de SpiderCatalog

Dentro de los Parámetros que podemos declarar, existe una combinación de Parámetros que son *type* y *resource*. Estos Parámetros nos permiten lanzar una función o aplicación que dependerá del tipo que sea (*type*) y del recurso incluido (*resource*). Los tipos de aplicaciones disponibles vienen especificados en la tabla 6.3.

Parámetros		
Tipo (<i>type</i>)	Recurso (<i>resource</i>)	Descripción
mail	Email info@mail.com	Lanza la aplicación para mandar un mail a la dirección en el recurso
phone	Nº de teléfono	Lanza una aplicación para hacer una llamada de teléfono al número del recurso
html	Código Html <!DOCTYPE HTML>	Lanza una ventana donde ejecuta un bloque de código html indicado en el recurso
pdf	Documento PDF tutorial.pdf	Lanza una ventana donde se muestra el pdf indicado en el recurso
url	Dirección URL www.___.com	Lanza una ventana donde muestra la dirección url contenida en el recurso
products	Interfaz de la aplicación	Lanza una ventana donde muestra los productos de un determinado formato mostrando información del producto
gallery	Interfaz de la aplicación	Lanza una ventana donde se muestra las subcategorías y productos en forma de lista

news	Noticias servidor RSS	Lanza una ventana donde mostrará las noticias del servidor RSS contenido en el recurso
-------------	-----------------------	--

Tabla 6.3 Parámetros de productos

6.1.1 Estructura de SpiderCatalog

Para empezar se debe dejar claro una serie de conceptos y el modelo sobre el que trabaja SpiderCatalog y que después se va a adaptar su funcionamiento para crear la aplicación a través del framework. La estructura con la que se va a trabajar siempre será una estructura en forma de árbol o ramificada como se muestra en la figura 6.1.

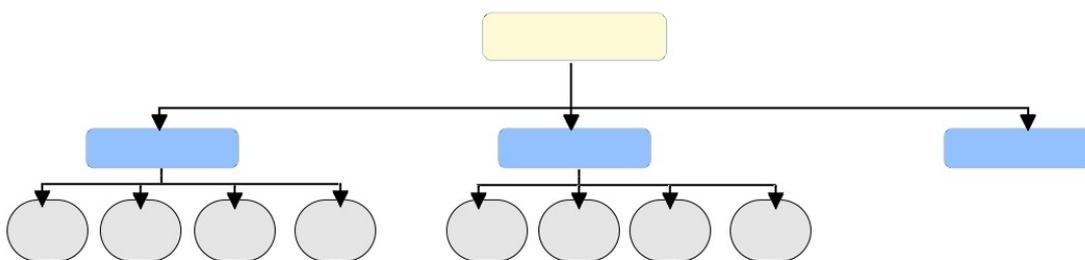


Figura 6.1 Estructura de SpiderCatalog

El modelo utilizado será categoría/Parámetro - producto/Valor. Así que dentro de nuestra estructura ramificada tendremos diferentes categorías y dentro de estas categorías tendremos diferentes productos y otras subcategorías que heredan las características de la categoría superior. Las categorías nos sirven para declarar Parámetros mientras que los productos, que pertenecerán a una o varias categorías, rellenarán con Valores los Parámetros de las categorías. Las subcategorías heredan los Parámetros de las categorías a las que pertenecen. Así pues estos dos elementos cumplen distintas funciones dentro de la estructura del sistema:

- **Categorías.** Cada categoría nos permite declarar un nombre de categoría,

una imagen asociada, la categoría a la que pertenece, de la cual heredará los parámetros, una descripción, los parámetros que deseemos añadir y el número de orden en el que aparecerá.

- **Productos.** Cada producto nos permite declarar un nombre de producto, una imagen asociada, la categoría a la que pertenece, de la cual heredará los parámetros, una descripción, los Valores de los Parámetros que le asignemos y el número de orden en el que aparecerá.

Reglas de diseño

Existen unas reglas de diseño que debemos respetar para poder llevar a cabo nuestro proyecto con éxito. Son las siguientes:

- Las categorías declaran los parámetros.
- Los productos declaran los valores de los parámetros.
- La categoría principal lleva como número de Ordering, el 1, pues este le otorga privilegios especiales y es así como se indica. Las demás llevarán numeración ordenada.
- Las subcategorías heredan los parámetros de las categorías a las que pertenecen.
- Pueden coexistir subcategorías con nombres similares, pero no dentro de la misma subcategoría.
- Es posible y aconsejable declarar un producto con el mismo nombre que la subcategoría a la que pertenece con el objetivo de asignar valores a sus parámetros y mostrarlos.
- Un mismo producto puede pertenecer a varias categorías, e incluso a distintos niveles de jerarquía.
- Dos categorías “hermanas” que comparten un mismo producto deben tener los mismos parámetros declarados en ambas categorías. Esto evita conflictos de programación a la hora de mostrar los datos.

6.2 Estructura del sistema

Definida la estructura con la que trabaja el *plugin*, se procede a realizar la estructura de categorías y productos con la que se va a definir la aplicación del sistema a crear.

Se parte de una categoría principal llamada Gran Canaria Cultural, ésta es la categoría inicial de la que parte la estructura del sistema, de esta categoría se ramifican el resto de subcategorías y productos que conforman el sistema.

De Gran Canaria Cultural se originan dos subcategorías llamadas Eventos y Redes Sociales. Con esta primera declaración ya se ha creado una estructura ramificada inicial con una categoría principal y dos subcategorías. Esta estructura está representada en la imagen 6.2

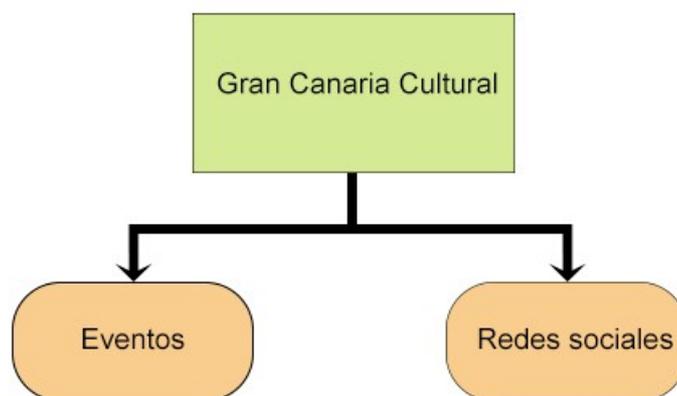


Figura 6.2 Estructura inicial del sistema

Para la creación de la estructura son imprescindibles los parámetros ID, Parent y Ordering. La categoría principal, que se ha denominado Gran Canaria Cultural, en estos parámetros tienen los siguientes valores que se muestran en la tabla 6.4.

Categoría Gran Canaria Cultural			
Name	ID	Parent	Ordering
Gran Canaria Cultural	1	0	1

Tabla 6.4 Categoría principal

Los valores 0 y 1 del campo Parent y Ordering respectivamente indican al framework que es la categoría inicial para crear la estructura del sistema y de la que se van a crear subcategorías y productos que heredarán de ésta.

Para la creación de las subcategorías Eventos y Redes Sociales hay que tener también en cuenta estos tres parámetros antes mencionados, con especial atención al valor Parent que tiene que tener el valor del ID de la categoría superior de la que hereda. En este caso el valor es 1 de Gran Canaria Cultural. El campo Ordering indica el orden en el que se muestran en la aplicación. El valor del campo ID tiene su importancia a la hora de crear subcategorías y/o productos de estas categorías. Todos estos valores están representados en la tabla 6.5.

Subcategorías de Gran Canaria Cultural			
Name	ID	Parent	Ordering
Eventos	2	1	2
Redes Sociales	40	1	4

Tabla 6.5 Subcategorías de la categoría principal

La categoría Eventos no contiene subcategorías de orden inferior, sí que contiene productos que son cada uno de los eventos creados en el sistema en WordPress y que se muestran tanto en el portal web como en la aplicación. La estructura del sistema con la inclusión de los eventos queda como se representa en la imagen 6.3

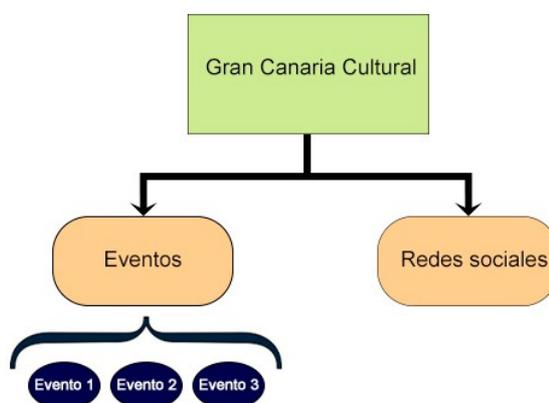


Figura 6.3 Estructura de Eventos

La categoría Redes Sociales contiene tres subcategorías que pertenecen a tres de las redes sociales más conocidas en la actualidad, Facebook, Twitter e Instagram. Cada una de estas subcategorías contiene un producto homónimo que redirecciona a cada una de la red social para que el usuario acceda al perfil del sistema en cada una de las redes sociales.

La estructura en árbol del sistema en la categoría de Redes Sociales queda como se muestra en la imagen 6.4 de la siguiente imagen. Ahí se comprueba como las subcategorías son ramificaciones de la categoría superior, y cada subcategoría tiene su producto de igual nombre.

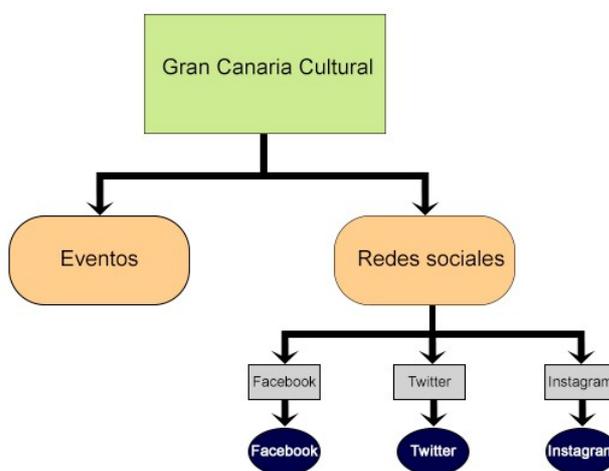


Figura 6.4 Estructura de Redes Sociales

6.2.1 Filtros

Los filtros ayudan a los usuarios a clasificar la totalidad del contenido incluido en el sistema para que su visualización sea más dinámica limitando el número de productos a mostrar en la aplicación.

Para esta aplicación se han creado dos tipos filtros para acotar el número de eventos según la localidad donde se realicen y la actividad que desarrollen. Estos filtros son:

- **Filtro por municipios.** Este filtro clasifica a los eventos según el lugar que tenga el evento.

- **Filtro por actividad.** Clasificación de los filtros según la actividad y al público al que va dirigido cada evento.
- **Creación de los filtros**

Para la creación de los filtros se sigue el modo de operación de las categorías y subcategorías, es decir, también tiene una estructura jerarquizada.

Primeramente hay que declarar las dos categorías principales en las que se va a clasificar la totalidad de los eventos del sistema. A estas categorías son asignados unos valores en sus parámetros como muestra la tabla 6.6, para que las categorías queden reflejadas como filtros.

Filtros			
Name	ID	Parent	Ordering
Filtrar por municipios	41	0	2
Filtrar por actividades	52	0	3

Tabla 6.6 Declaración para la creación de un filtro

Con estos valores asignados tiene las propiedades para ser una categoría en el nivel más alto de la estructura árbol al ser con un *Parent* de valor 0 pero con cualidades para hacer de filtro.

Después de declarar las categorías principales de los filtros se procede a la creación de los filtros correspondientes a cada uno de los municipios y actividades que filtran los eventos. Para crearlas se sigue el mismo procedimiento que con las categorías y las subcategorías. Se comprueba en la figura 6.5 que los filtros no son una ramificación de la categoría principal, están en el mismo nivel ya que tienen *Parent* de valor 0.

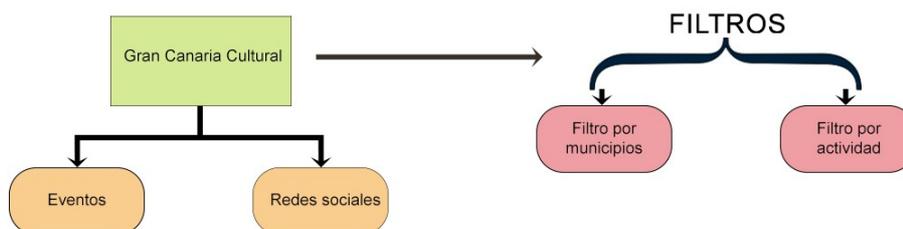


Figura 6.5 Estructura con los filtros

Las subcategorías de Filtros por municipios está creada para dividir el sistema en los diferentes municipios que conforman la isla de Gran Canaria. Esto quiere decir que dentro de la categoría municipios existen veintidós subcategorías hijas, cada una correspondiente a cada uno de los municipios.

Las subcategorías de Filtros por actividades dividen al sistema por la actividad que desarrolla cada evento. Éstas son: deportes, fiestas populares, charlas y conferencias, conciertos musicales, espectáculos, infantil, exposiciones y varios.

Como cada declaración de subcategorías hay que identificarlas con su categoría de orden superior dentro de la estructura. Para ello se le asigna el valor al campo Parent similar al campo ID de la categoría. La tabla 6.7 muestra los valores que tienen los filtros de municipios.

Filtros de municipios			
	ID	Parent	Ordering
Agaete	4	41	2
Agüimes	5	41	3
Artenara	6	41	4
Arucas	7	41	5
Firgas	8	41	6
Gáldar	9	41	7
Ingenio	10	41	8
La Aldea de San Nicolás	11	41	9
Las Palmas de G.C.	12	41	10
Mogán	13	41	11
Moya	14	41	12
San Bartolomé de Tirajana	15	41	13
Santa Brígida	16	41	14
Santa Lucía de Tirajana	17	41	15
Santa María de Guía	18	41	16

Tejeda	19	41	17
Telde	20	41	18
Teror	21	41	19
Valleseco	22	41	20
Valsequillo	23	41	21
Vega de San Mateo	24	41	22

Tabla 6.7 Filtros de Municipios

En la tabla anterior se comprueba que Parent tiene valor 41 como el ID de la categoría Municipios, y en la tabla siguiente tiene valor 52 como el ID de Actividades. En ambas se muestran los valores de ID que identifican a las categorías y el valor Ordering que indica el orden en el que se muestran las categorías. La tabla 6.8 muestra los valores de los filtros por actividades.

Filtros de actividades			
	ID	Parent	Ordering
Deportes	25	52	2
Fiestas Populares	26	52	3
Charlas y Conferencias	27	52	4
Espectáculos	29	52	6
Infantil	30	52	7
Exposiciones	31	52	8
Varios	32	52	9

Tabla 6.8 Subcategorías de categorías

Por lo que la estructura ramificada del sistema completo incluyendo la categoría principal y sus subcategorías y los filtros queda resumida de la forma que muestra la siguiente imagen 6.6 que se muestra a continuación.

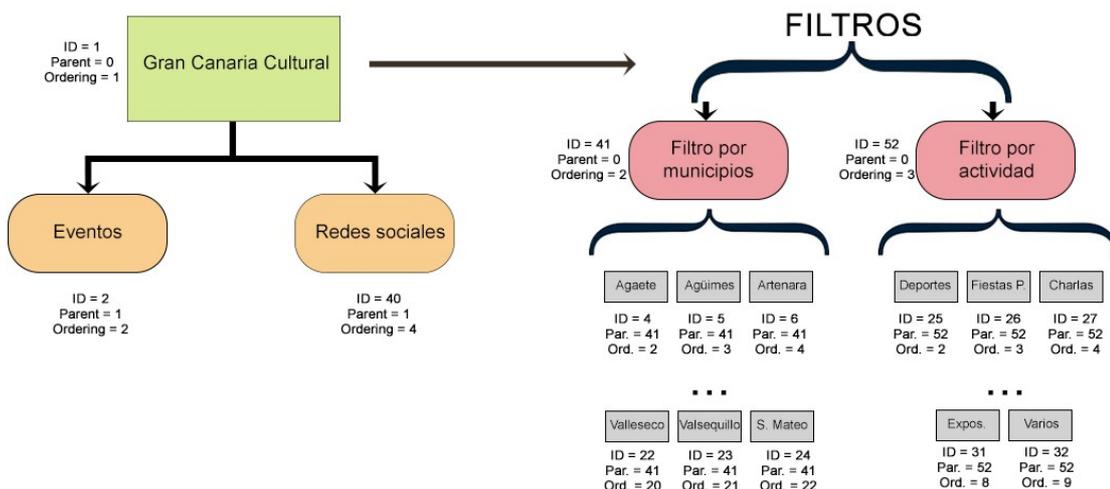


Figura 6.6 Estructura del Sistema

Los productos dentro del sistema

Los productos principales del sistema son los eventos organizados por los organismos oficiales de Gran Canaria. Todos estos productos siempre tienen al menos una localización y desarrollan un tipo de actividad. Por lo que todos pertenecen siempre a alguno de los filtros antes declarados. Además todos los eventos por defecto se agrupan dentro de la categoría Eventos, que como se describió anteriormente aglutina todos los eventos del sistema.

Si se declara un evento con las siguientes características que muestra la tabla 6.9:

Producto 1		
Name	Category_id	Ordering
Evento 1	2,4,27,31	1

Tabla 6.9 Declaración de un producto

Los valores del atributo Category_id indica a que categorías pertenece el evento declarado Evento1, los valores de este atributo indican que efectivamente se incluye dentro de la categoría Eventos, pertenece a la categoría Agaete dentro del filtro de municipios y a las categorías Charlas y conferencias y Exposiciones dentro del filtro de actividades. En la figura 6.7 se muestra como se organiza

dentro de la estructura la creación de un producto, en este caso Evento 1.

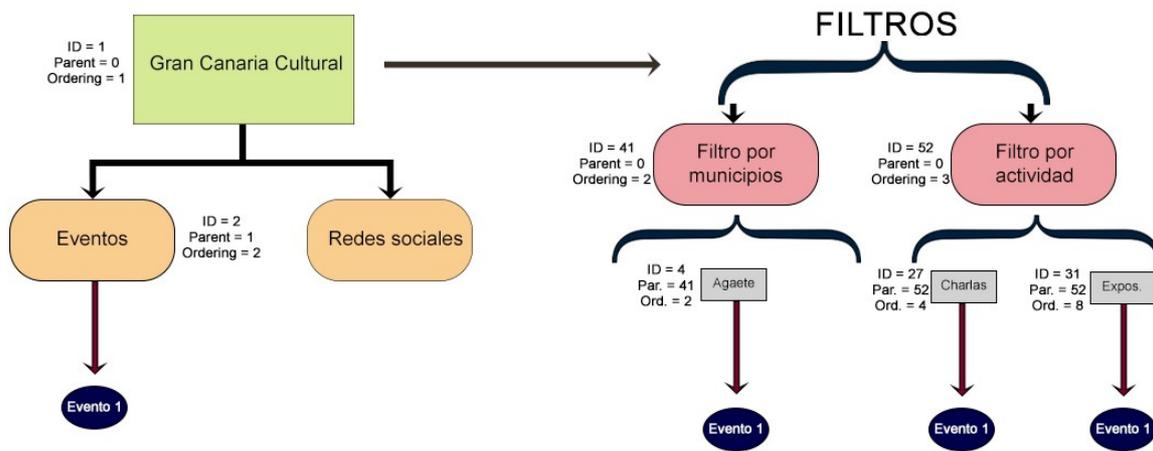


Figura 6.7 Estructura con un evento

6.3 Adaptación del sistema a la estructura de SpiderCatalog

La realización del trabajo realizado para crear un entorno y estructura similar al plugin de WordPress se divide en dos tareas: La creación de categorías desde Parse y la creación de productos que se forman obteniendo los valores desde las entradas creadas para los eventos en WordPress y almacenadas en otra tabla de Parse. Estando los datos almacenados en Parse son accesibles al framework para crear la aplicación.

- **Creación de las categorías**

El conjunto de categorías y filtros que componen el sistema se crean directamente accediendo al portal de Parse.

Se añade una tabla denominada Categories con los campos necesarios para guardar los parámetros que definen las categorías. Estos campos son Name, Description, Image_url, Parent, Param, ID, Ordering y Published. La imagen 6.8

muestra una imagen de la tabla Categories en Parse.

category_image_url String	description String	name String	ordering	param String	parent String	published ...	spider_id ...
http://grancanariacultural.s3.amazonaws...		GC Cultural	1	type=resourcesubtypeactionstrongColormedium...	0	1	1

Figura 6.8 Tabla Categories

Se crean las categorías antes declaradas con los valores indicados para que tengan la estructura en forma de árbol necesaria como se ha indicado en la Figura 6.7. Siendo Gran Canaria Cultural la categoría principal de la que se van añadiendo subcategorías hasta formar la estructura completa de categorías y subcategorías. Y de forma paralela la creación de los filtros que conforman el sistema.

- **Creación de los productos**

Los productos se almacenan en otra tabla creada en Parse llamada *Products*. Esta tabla al igual que la tabla creada para las categorías tiene tantos campos como parámetros tienen los productos. Estos campos son Name, Description, Image_url, Param, Category_Id, Ordering y Published.

Los productos almacenan los valores de los eventos creados en el sistema, estos valores se obtienen de las entradas creadas en WordPress. Cada línea nueva creada en la tabla de Products de Parse pertenece a un nuevo evento creado en WordPress. La figura 6.9 muestra a modo de ejemplo como queda una línea de la tabla Products cuando se almacenan los datos de un evento.

spider_id String	category_id String	ordering String	name String	description String
275	12,31,44,	275	Ciclo 'El Greco 400 años después. Una visión contemporánea'	<p>Ciclo 'El Greco 400 años después. Una visión contempor

Figura 6.9 Tabla Products

Los datos creados en las entradas de WordPress se almacenan en variables PHP y son enviadas en formato JSON hacia Parse para una vez allí ser almacenados en la tabla Products. Todo este proceso se realiza de forma automática una vez se crea el evento en WordPress. La tabla 6.10 muestra que a cada campo de la tabla Products en Parse le corresponde una campo del editor

de entradas de WordPress.

Asignación de valores a productos	
Parse	Editor de WordPress
Name	Título del evento
Description	Meta box creados para almacenar la fecha, hora y descripción del evento
Image_url	Url de la imagen previamente guardada en Amazon
Category_Id	Categorías de WordPress y Meta box tipos de eventos que indica a que categorías pertenece el evento
Ordering	Id de la entrada en WordPress

Tabla 6.10 Valores de Products

Al publicar una nueva entrada en WordPress automáticamente se almacenan los datos en Parse creando un fila más en la tabla Products con los campos provenientes de la nueva entrada en WordPress y son leídos por la el framework para que sean visibles en la aplicación.

6.3.1 Configuraciones de la app

Imágenes

Se han creado varios tipos de imágenes para para incluir en la aplicación para crear una interfaz de usuario más amigable y que la aplicación no contenga únicamente texto. Existen tres tipos de imágenes que se incluyen en la aplicación:

- Las imágenes que se incluyen en los eventos. Este tipo de imágenes se incluyen a través de WordPress y normalmente son los carteles que se crean para anunciar dichos eventos. Estas imágenes pueden tener diferentes formatos y el *framework* las adapta para mostrarlas de la manera más óptima en los dispositivos móviles.
- La imagen del logotipo del sistema. Esta imagen creada para el proyecto y que se muestra en el menú principal de la aplicación debe ser una imagen

cuadrada. La imagen creada tiene un formato de 350x350 píxeles. También es utilizada esta imagen como icono de aplicaciones.

- Imágenes que se presentan al inicio de los menús, en el caso de la aplicación creada para el sistema del proyecto es la imagen de la subcategoría Redes Sociales (figura 6.13). Esta categoría de imágenes deben de tener un formato de 500 x 200 píxeles por lo que han tenido que ser modificadas con el programa de edición de imágenes GIMP.

Notificaciones Push

La tecnología push es un tipo de comunicación donde es el servidor el que inicia la conversación en dirección al cliente. El proceder habitual en las aplicaciones era la tecnología *pull*, ésta se caracteriza por ser el cliente el que realiza la petición al servidor pidiendo una actualización de los datos.

Las ventajas de las notificaciones push con respecto a la tecnología pull son:

- Ahorro de los recursos del móvil. Ya que no es necesario que sea la aplicación la que se active para buscar actualización de los datos, sino que el servidor avisa cuando hay nuevos datos. Al no tener la aplicación activa también supone un ahorro de la batería del móvil.
- La notificación de los nuevos datos es instantánea. Los datos están accesibles en el móvil desde que son enviados por el servidor.

La tecnología push está incluida en el sistema para la notificación de nuevos eventos publicados. Esto quiere decir que en el momento que un evento es creado por un editor de contenidos en WordPress y es almacenado en Parse se envía una notificación push a los móviles de los usuarios.

Las notificaciones push no se envían a todos los usuarios de la aplicación, para recibir estas notificaciones un usuario debe indicar dentro de la aplicación los tipos de eventos de los que quiere ser informado por este método. Un usuario se

puede suscribir al envío de estas notificaciones por municipios y/o categorías, es decir, un usuario puede indicar al sistema que quiere recibir notificaciones push de 2 ayuntamientos en concreto y de 3 categorías indistintamente a los municipios que pertenezcan.

Para suscribirse a las notificaciones el usuario tiene que activar el filtro de la categoría de la quiere recibir las notificaciones, una vez activado el filtro cada nuevo evento que se incorpore al sistema que sea de la categoría filtrada se notificará al usuario mediante una notificación push.

Para crear notificaciones push en el sistema hay que incluir un atributo dentro de los productos correspondientes a las distintas subcategorías de los filtros de municipios y actividades, es decir, a cada uno de los 21 ayuntamientos y los 7 tipos de eventos. El atributo a modificar es *action*. La tabla 6.11 muestra el valor del parámetro *action*.

Parámetro		
atributo	tipo	Descripción
action	notificable	Indica al sistema que la categoría con el mismo nombre envía notificaciones push.

Tabla 6.11 Parámetro para la notificación push

Con esta acción se indica al sistema que se va a crear un canal para enviar notificaciones push de los eventos de esa categoría a los usuarios que hayan demandado recibir las notificaciones push.

6.4 Ejemplos

A través de los siguientes ejemplos se expone los efectos que tienen las decisiones tomadas a la hora de diseñar el sistema.

Se muestran como quedan organizadas las categorías, los filtros y los productos en la aplicación, y también se ve como afectan los distintos parámetros declarados y cómo la aplicación los muestra al usuario.

La Figura 6.10 muestra la pantalla inicial al abrir la aplicación, el menú principal y los eventos que componen todo el sistema. En la Figura 6.11 se señala las diferentes categorías y productos que se han declarado en Parse que conforman el *slide* menú y como el framework recoge los valores de los parámetros de la categoría principal Gran Canaria Cultural para completar el menú.

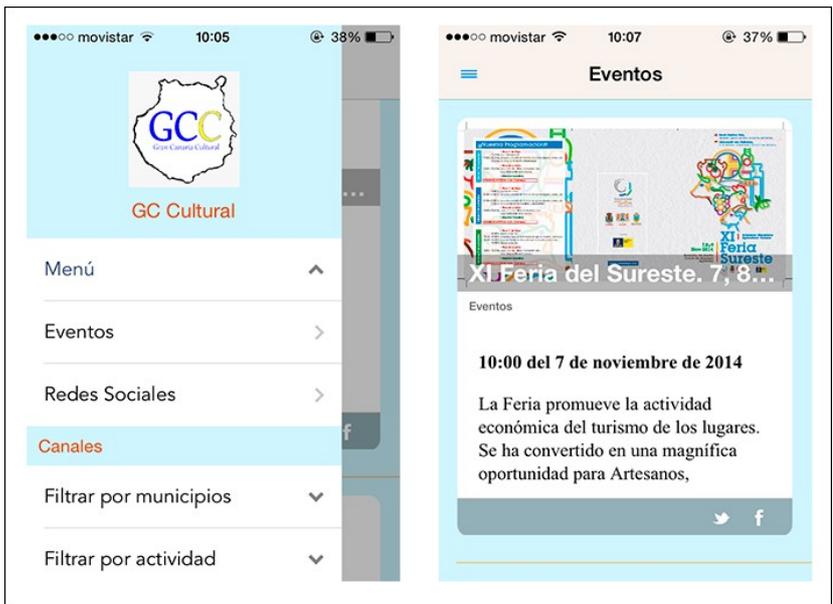


Figura 6.10 Pantalla inicial

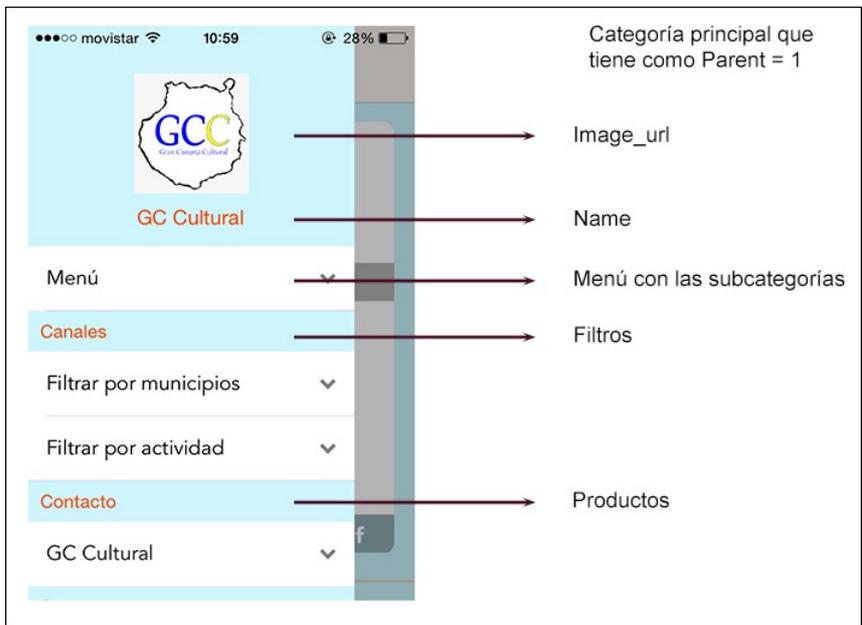


Figura 6.11 Slide menú

La figura 6.12 muestra el *slide* menú. El slide menú es desplegable para mostrar el menú principal dependiente de la categoría inicial y los filtros que se aplican para acotar el número de eventos a mostrar al usuario.

En la Figura 6.13 se muestra la composición de la subcategoría Redes Sociales, en la que aparece el listado de sus subcategorías, el sistema carga las imágenes dentro del campo "Image_url" de Municipios e informa de la existencia de información debido a un producto de Ordering 1.



Figura 6.12 Categorías y Filtros



Figura 6.13 Diseño de una categoría

La siguiente imagen, figura 6.14, expone la acción de acceder a la información contenida en un producto de Ordering 1 que muestra la información incluida dentro del campo “description”. La Figura 6.15 muestra como los productos pertenecientes a la categoría principal también son ubicados en el slide menú.



Figura 6.14 Información de Ordering 1



Figura 6.15 Productos de la categoría Principal

Los tipos de productos definen la interfaz de usuario que se activa en la aplicación. En la aplicación creada para el proyecto fin de carrera se han utilizado los tipos de producto gallery, products, news, mail y url. A continuación se muestran imágenes de todos los tipos de productos usados. La figura 6.16 muestra como se representa en el móvil cuando un producto es de tipo *gallery*.



Figura 6.16 Tipo gallery

La figura 6.17 muestra la representación de un producto tipo *products*.



Figura 6.17 Tipo products

El tipo *news* se representa en la aplicación como muestra en la imagen 6.18.



Figura 6.18 Tipo news

La figura 6.19 muestra la pantalla del producto tipo *mail* para enviar un correo electrónico.



Figura 6.19 Tipo mail

El tipo *url* de la figura 6.20 muestra como la aplicación redirecciona a la url indicada.



Figura 6.20 Tipo url

Conclusiones

El proceso de realización del Proyecto Fin de Carrera tenía como objetivo principal del proyecto la realización de una aplicación móvil para la difusión de eventos culturales de la isla de Gran Canaria.

De forma más detallada se indican en la tabla 7.1 los diferentes objetivos que se pretendían realizar para llevar a cabo esta tarea.

Objetivo	Detalles
Aprendizaje de nuevos lenguajes de programación orientados a web y aplicaciones móvil	Se ha aprendido nuevos lenguajes como son HTML, Javascript, PHP, JSON, etc. Ampliando los conocimientos de programación aprendidos en la carrera
Estudio y uso de los gestores de contenidos para la creación de portales web	Experiencia en el mundo de los CMS, en especial de WordPress, que me otorga la capacidad para crear portales web.
Iniciarse en el patrón modelo-vista-controlador	Iniciación al patrón MVC y experiencia al trabajar con un framework orientado a trabajar con él.
Introducción a los servicios en la nube	Descubrimiento de los servicios en la nube que amplían las posibilidades para la creación de aplicaciones web y móviles.

Tabla 7.1 Objetivos del PFC

Analizando estos objetivos y viendo que se han cumplido todos los objetivos marcados se obtiene como conclusión final, que se ha conseguido crear la aplicación móvil marcada como objetivo principal y se han ampliado los conocimientos sobre otras herramientas y métodos de trabajo aplicando los conocimientos adquiridos a lo largo de la carrera de ingeniería técnica de telecomunicación, en la especialidad de telemática.

Bibliografía

- [1] *“WordPress, Joomla!, Drupal, ¿cuál es el mejor gestor de contenidos (CMS)?”*.
<http://mongemalo.es/wordpress-joomla-drupal-mejor-gestor-contenidos-cms/>
Último acceso a la web el 12 de Septiembre de 2014.
- [2] *“Statistics for websites using CMS technologies”*.
<http://trends.builtwith.com/cms>
Último acceso a la web el 23 de Septiembre de 2014.
- [3] *“Página oficial de Wopzen2”*.
<https://wordpress.org/plugins/wopzen2/>
Último acceso a la web el 10 de Junio de 2014.
- [4] Encuesta sobre el uso de Tecnologías de la Información y las Comunicaciones (TIC) y del comercio electrónico en las empresas
<http://www.ine.es/prensa/np859.pdf>
- [5] *No me hagas pensar. Una aproximación a la usabilidad web*.
Steve Krug. Prentice-hall, 2006.
- [6] *Diseño web adaptativo. Responsive web design*. Pedro Clemente Bonilla.
Editorial Anaya Multimedia, 2013.
- [7] *“Portal web que compara la cuota de mercado de los navegadores”*.
www.statcounter.com
Último acceso a la web el 20 de Septiembre de 2014.
- [8] *“Portal web que compara la cuota de mercado de los navegadores”*.
www.netmarketshare.com
Último acceso a la web el 20 de Septiembre de 2014.

- [9] *“Página oficial de XAMPP”*.
<http://www.apachefriends.org/es/xampp.html>
Último acceso a la web el 14 de Mayo de 2014.
- [10] *“Página oficial de WordPress”*.
<http://wordpress.org/>
Último acceso a la web el 16 de Julio de 2014.
- [11] *“Página oficial de Composer”*.
<https://getcomposer.org>
Último acceso a la web el 8 de Mayo de 2014.
- [12] *“Guía para la instalación de WordPress en Heroku”*.
<http://www.andrewgertig.com/2013/05/set-up-wordpress-on-heroku>
Último acceso a la web el 28 de Octubre de 2014.
- [13] *Diseño Web : edición 2008*. Jason Beard. Editorial Anaya, 2008.
- [14] *El gran libro de HTML5, CSS3 & Javascript*. Juan diego Gauchat. Editorial Marcombo, 2013.
- [15] *¿Qué es la nube?: el futuro de los sistemas de información*. Barrie Sosinsky. Editorial Anaya, 2011.

Pliego de Condiciones

Pliego de condiciones

1.1 Lenguajes de programación

1.1.1 HTML

HTML, HyperText Markup Language, es un sencillo lenguaje utilizado, que no creado, para la creación de portales web. Es un lenguaje hipertexto, esto quiere decir que no es un lenguaje secuencial y que no se compone únicamente de texto, también puede compartir información de diversas fuentes como imágenes, sonidos, etc.

Para escribir código HTML no se precisa ningún editor especial ya que cualquier editor de texto ASCII es totalmente válido. HTML se escribe en forma de «etiquetas», rodeadas por corchetes angulares (<etiqueta>contenido</etiqueta>). Existen diversas etiquetas según lo que se quiera incluir dentro del documento, ya que no es igual una etiqueta que nos muestre el título del documento a otra para incluir una imagen en un texto.

Todo documento HTML comienza con una marca de inicio <html> y finaliza con otra de cierre </html>. Estas marcas indican al navegador que el documento está escrito en HTML. La estructura de un documento HTML tiene dos partes bien diferenciadas:

- **Cabecera.** Empieza con la marca <head> y finaliza con </head>. La información incluida en la cabecera, no se muestra al usuario pero sí que contiene importante información para el correcto funcionamiento y visualización de un página HTML en un navegador web.
- **Cuerpo.** Comienza con la marca <body> y finaliza con </body>. Dentro del cuerpo del documento es donde reside la información que se muestra al usuario.

La imagen 1.1 muestra estos tres elementos que forman la estructura

básica del código HTML.

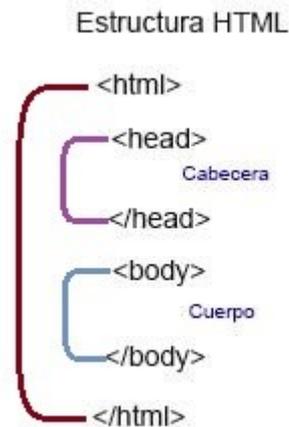


Figura 1.1 Estructura básica de HTML

Existe gran cantidad de etiquetas en el código HTML, a continuación se muestran algunas de las etiquetas más utilizadas para dar forma a un documento HTML:

- **<div>**: Probablemente sea el elemento más importante dentro del HTML. Este elemento nos permiten agrupar un conjunto de elementos y aplicar reglas de estilo.
- **<p>**: Se utilizan para separar el texto escrito en párrafos. Ya que el texto es continuo en HTML se necesita algún mecanismo que indique cuando hay un cambio de párrafo.
- **<hx>**: Es un formato predefinido utilizado regularmente para indicar los títulos de un texto. X es una variable cuyo valor varía entre 1 y 6 para variar el tamaño de la fuente.
- **<a>**: Marca de hipervínculo. Un navegador al encontrar esta marca resalta el texto y al hacer clic sobre ella nos permiten cargar otra página en el navegador.
- **<form>**: Indica el inicio de un formulario. Los formularios son muy utilizados para obtener información del usuario.

Todas estas etiquetas son declaradas dentro del cuerpo del programa y contribuyen a crear un documento HTML más enriquecido mostrando un mejor contenido.

Los navegadores son los encargados de interpretar el lenguaje HTML y de mostrar el resultado a los usuarios de la página web. Éste es uno de los grandes problemas con los que se encuentra un programador porque hay gran cantidad de navegadores en el mercado y no son capaces de interpretar un mismo código de una manera unificada.

Otro de los problemas de HTML es que no fue creado inicialmente para la creación de páginas web ni para dar soporte a todos los usos que finalmente se han ido incluyendo. Para dar solución se han tenido que ir incorporando otras tecnologías complementarias como CSS y Javascript e ir creando nuevas modificaciones en estándares para ir adaptando todos esos posibles usos. El último es el estándar en el que se está trabajando, el HTML5.

- **HTML5**

HTML5 es la última evolución de la norma que define HTML y viene a sustituir a HTML4, XHTML y DOM HTML. La estandarización oficial y la especificación de HTML5 ha sido aprobada recientemente, en octubre de 2014.

En esta versión se incluyen nuevas etiquetas y nuevos atributos para utilizar en nuestro documento web, pero además se ha definido una manera de cómo estructurar el contenido de nuestro documento. HTML5 proporciona una serie de recursos y facilidades para ayudar a desarrollar aplicaciones web sin necesidad de ayuda externa más allá del lenguaje javascript en muchos casos, ésto hace que se reduzca la utilización de plugins como el Adobe Flash.

1.1.2 CSS

CSS (Cascade Style Sheet) hojas de estilo en cascada, es un lenguaje que describe la presentación de los documentos estructurados en hojas de estilo para diferentes métodos de interpretación, es decir, describe como se va a mostrar un

documento.

Las hojas de estilo se desarrollaron para compensar los defectos de HTML con respecto a la presentación y al diseño de las páginas. HTML tiene varias etiquetas para modificar la presentación y definir los estilos del texto, pero cada elemento tiene su propio estilo, independientemente de los elementos que lo rodean. Al utilizar hojas de estilo, cuando se necesite cambiar la apariencia de un sitio que tienen varias páginas web todo lo que hay que hacer es editar las definiciones de la hoja de estilo en un solo lugar para cambiar la apariencia del sitio completo. Se denominan "hojas de estilo en cascada" porque se pueden definir múltiples hojas y los estilos pueden aplicarse a todas las páginas (con un sistema predefinido para resolver conflictos).

El lenguaje CSS apareció por primera vez en 1996 y está definido en la especificaciones CSS1, CSS2 y CSS3 del *World Wide Web Consortium* (W3C) y es un estándar aceptado por toda la industria relacionada con la Web.

Utilizando las hojas de estilo podemos realizar multitud de variaciones a una página HTML. Se puede cambiar el color y el tamaño del texto, incluir bordes a una imagen o indicar la posición de cada elemento dentro de la página.

1.1.3 Javascript

JavaScript es un lenguaje de programación interpretado. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico, y es utilizado para acceder a objetos en aplicaciones.

Javascript nació con la necesidad de permitir a los autores de sitio web crear páginas que permitan intercambiar con los usuarios, ya que se necesitaba crear webs de mayor complejidad. El HTML sólo permitía crear páginas estáticas donde se podía mostrar textos con estilos, pero se necesitaba interactuar con los usuarios.

Se utiliza principalmente en su forma del lado del cliente, técnicamente llamado Navigator JavaScript, por lo que no requiere compilación, implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas. Las sentencias JavaScript embebidas en un documento

HTML pueden reconocer y responder a eventos generados por el usuario, como clics del ratón o interpretando información en formularios comprobando que ésta sea correcta y alertar al usuario en caso de que no lo sea.

JavaScript se diseñó con una sintaxis similar al C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo Java y JavaScript no están relacionados y tienen semánticas y propósitos diferentes.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del *Document Object Model* (DOM).

1.1.4 PHP

PHP, acrónimo recursivo de *Hypertext Pre-processor*, es un lenguaje interpretado de código abierto y multiplataforma especializado para el desarrollo web en el lado del servidor, por lo que el cliente no ve el código sino los resultados que éste produce, y que puede ser fácilmente incrustado en el código HTML.

La primera versión de PHP fue creada en 1994 por Rasmus Lerdorf y estaba basado en la escritura de un conjunto de CGI del lenguaje C. Ya en 1997 Zeev Suraski y Andi Gutmans programaron el analizador sintáctico incluyendo nuevas funcionalidades como el soporte a nuevos protocolos de Internet y el soporte a la gran mayoría de las bases de datos comerciales, como MySQL y Postgre SQL, así como un módulo para Apache. Actualmente se trabaja con la versión PHP5 que proporciona mejoras significativas y un entorno de programación orientado a objetos mucho más completo.

Existen otros lenguajes similares como ASP o JSP, pero PHP es el más utilizado por la mayoría de desarrolladores web debido a que es Open Source (código abierto), de fácil aprendizaje para principiantes debido a su similitud con lenguajes de programación de tipo estructurado como C y a la vez ofrece características avanzadas para programadores más expertos.

PHP permite el acceso a acceso a las bases de datos y a otras funcionalidades orientadas a la red. Pero la característica más importante de PHP

es que permite modificar dinámicamente el contenido de una página, lo cual resulta esencial hoy en día. El cliente hace una petición al servidor para que le envíe una página web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado que generará el contenido de manera dinámica. El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente. La imagen 1.2 muestra el esquema de funcionamiento de PHP al realizar una petición a la base de datos.

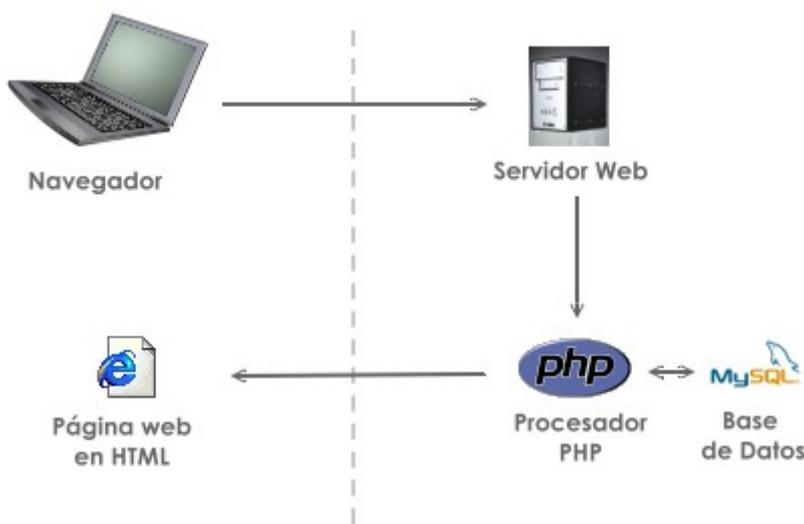


Figura 1.2 Esquema de funcionamiento de PHP

1.1.5 JSON

JSON (JavaScript Object Notation - Notación de Objetos de JavaScript) es un formato ligero de intercambio de datos. Es un formato de texto que es completamente independiente del lenguaje pero utiliza convenciones que son ampliamente conocidos por los programadores de la familia de lenguajes C (C, C++, Javascript, etc.). Ésta es una de las mayores ventajas que tiene el uso de JSON ya que al ser leído por cualquier lenguaje de programación puede ser usado para el intercambio de información entre distintas tecnologías.

JSON está constituido por dos estructuras:

- Una colección de pares nombre / valor. Conocido como objeto o registro.

Se conforma de una llave de apertura, el nombre del objeto entrecomillado, dos puntos, el valor dado al objeto y una llave de cierre.

```
{ "objeto" : valor }
```

- Una lista ordenada de valores. Una colección de valores que se declaran al inicio y al final con apertura y cierre corchete ([]) y los valores que contienen se separan por comas.

```
"datos": [{"objeto1" : "valor string", "objeto": valor int, "objeto2": "valor3"}]
```

La utilización de los lenguajes antes descritos combinados entre sí ha formado el código de programación creado para este proyecto fin de carrera. La creación de código ha sido realizado para crear páginas en WordPress, formularios dentro de Zend Framework, conexión para la comunicación de WordPress con Parse, etc.

1.2 Herramientas utilizadas

1.2.1 XAMPP

XAMPP es un servidor independiente de plataforma que consiste en un paquete formado por un servidor web Apache, una base de datos MySQL y los intérpretes para los lenguajes PHP y Perl. De hecho su nombre viene de ahí, X (para cualquier sistema operativo), A (Apache), M (MySQL), P (PHP) y P (Perl). XAMPP es independiente de plataforma y tiene licencia GNU GPL. Una de las ventajas de XAMPP es que de una forma muy sencilla y rápida un usuario puede montar en su ordenador un entorno de desarrollo de cualquier aplicación web que use PHP y base de datos.

La versión de XAMPP utilizada es la 1.8.2, la más actual en el momento de instalar el servidor para comenzar a trabajar. Esta versión para el sistema operativo Windows 7 se compone de: Apache 2.4.9, MySQL 5.5.36, PHP 5.4.27 por lo que cumple con lo necesario para su instalación.

XAMPP se ha utilizado en este proyecto para montar un servidor en el ordenador de trabajo para hacer todo el trabajo previo pudiendo hacer de una

manera más inmediata la validación y modificación de errores.

1.2.2 PHPStorm

PhpStorm es un entorno de desarrollo (IDE) utilizado para la construcción y edición de sitios y aplicaciones web. Permite editar código de los lenguajes que van a ser utilizados para el proyecto como HTML, PHP o CSS.

Los editores de código fuente tienen características diseñadas exclusivamente para simplificar y acelerar la escritura de código fuente, como resaltado de sintaxis, autocompletar y pareo de llaves.

Este editor no es de código libre pero se ha utilizado en este proyecto debido a que la Universidad de Las Palmas de Gran Canaria tiene licencia para su utilización para este tipo de proyectos. En él hemos creado y modificado todo el código necesario para llevar a cabo este proyecto.

1.2.3 Balsamiq mockup

Balsamiq Mockup es un programa para la realización de bocetos de aplicaciones móviles y de escritorio cuya finalidad es ayudar al desarrollo de aplicaciones con una herramienta que facilita la creación de esquemas.

Su idea es la de sustituir a los bocetos hechos en papel y para ello tiene una interfaz de usuario en los que imita a estos trazos y una facilidad y sencillez que permite hacerlo a la misma velocidad.

Se ha utilizado Balsamiq para crear una primera idea sobre lo que queríamos plantear como portal web. Tiene una versión de pago que no nos ha sido necesaria ya que tiene una versión menor y gratuita que contiene los elementos necesarios para nuestro trabajo.

1.2.4 GIMP

El programa GIMP (GNU Image Manipulation Program) es un software dedicado a la edición de imágenes. Éste al ser un programa libre, gratuito y

multiplataforma es el elegido para la creación y edición de las imágenes utilizadas en el proyecto.

GIMP permite a los desarrolladores editar y retocar imágenes propias pudiendo realizar fotomontajes, añadiendo iconos o texto.

En este proyecto se ha utilizado GIMP para crear y editar las imágenes utilizadas como por ejemplo el logotipo creado para la web y las aplicaciones móviles o las imágenes de los distintos municipios dentro de la sección agenda dentro del portal web.

1.2.5 GIT

Git es un sistema de control de versiones (SCM) creado por Linus Torvalds, creador de Linux. Este tipo de sistema, como el nombre sugiere, permite llevar un control de revisiones de documentos dentro de un proyecto. Así cada vez que un archivo cambia, este sistema se da cuenta de esos cambios y genera una nueva versión. Ésto permite mantener un historial de todos los cambios que se han hecho y de ser necesario llevar a un archivo o todo el proyecto a un estado anterior. Los SCM suponen un avance a la hora de coordinar los cambios que realiza un conjunto de desarrolladores sobre los ficheros de un proyecto.

Git es un SCM distribuido, que quiere decir que cada usuario tiene su propio repositorio de todo el código (un repositorio es el lugar donde se almacenan los datos tanto actuales como históricos). Los cambios que cada usuario haga del código dentro de su repositorio no afectan a los demás. Luego el desarrollador principal coge el código de los repositorios, lo combina y obtiene el código final. Todo ésto siempre hablando de proyectos en grupo.

- **GitHub**

GitHub es una plataforma de desarrollo colaborativo de software para alojar proyectos utilizando el sistema Git. Los usuarios pueden subir a GitHub todos los proyectos que quieran gratuitamente siempre que sean de carácter público. También existe la posibilidad de tener proyectos privados creando una cuenta de

pago.

Todo proyecto bajo el control de Git puede ser alojado en Github haciendo push (subiendo el código) desde tu repositorio en local hacia github creando allí una réplica. Cada vez que se envía código se realiza un *commit*, que sucede cuando una copia de los cambios hechos a una copia local se almacena en Github.

Para el manejo de proyectos con la plataforma Heroku se deben realizar a través de Git. En el momento de subir la aplicación a la plataforma o en el cambio de código se debe hacer a través de los commits.

1.3 INSTALACIÓN DE SERVICIOS

1.3.1 Instalación de un servidor local

La instalación de un servidor local, en el ordenador de trabajo, ha sido necesaria para el estudio inicial del gestor de contenidos WordPress y el framework Zend Framework 2. Ambos desarrolladores se han instalado de forma local para poder trabajar en el aprendizaje y familiarización de estos entornos y una vez visto el funcionamiento de éstos para realizar todas las pruebas oportunas dentro de éstos para la creación del portal web.

- **Requisitos de instalación.**

Para la instalación de WordPress 3.8, que era la versión más reciente en el momento de la instalación, se necesitan estos requisitos mínimos:

- **Intérprete y lenguaje**, la versión de PHP 5.2.4 o superior .
- **Servidor de bases de datos**, MySQL versión 5.0 o superior.
- **Servidor http**, Apache 2.2 o Nginx.

Para tener todos estos componentes instalados en el servidor local se procede a la instalación del programa XAMPP. Posiblemente XAMPP sea el más

transparente, sencillo y extendido programa para personas que se inician en este campo, que nos ofrece los requisitos necesarios para instalar WordPress.

- **Instalación de XAMPP**

La instalación de XAMPP es una tarea que no tiene excesiva dificultad para un usuario medio tecnológicamente hablando ya que es similar a la instalación habitual de software en un ordenador. Se puede clasificar esta tarea en 3 claros pasos: descargar, instalar y probar.

Descargar

La descarga del programa se realiza conectándose a la web oficial de XAMPP.[9]

En la web se encontrará todas las versiones actuales de XAMPP correspondientes a los distintos sistemas operativos con los que puede trabajar. Seleccionamos el correspondiente al nuestro, lo descargamos y lo guardamos en nuestro ordenador. La figura 1.3 muestra la web oficial de XAMPP desde donde se descarga el programa.

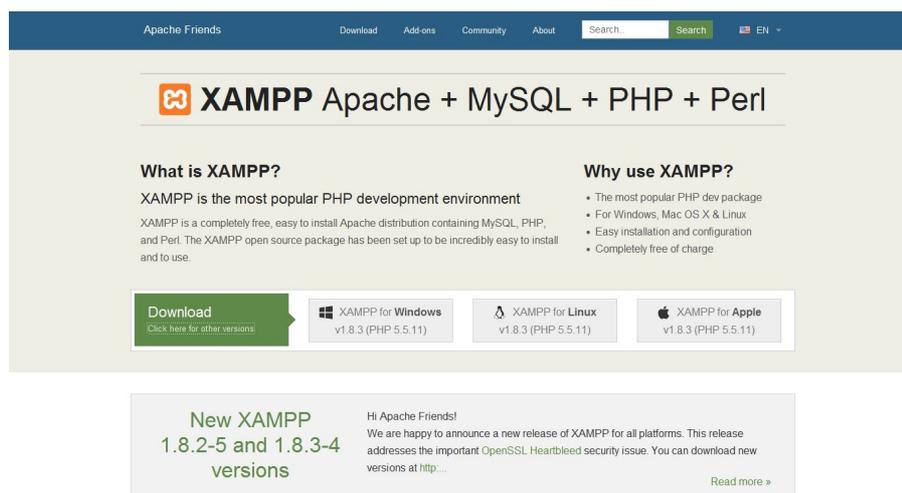


Figura 1.3 Web oficial de XAMPP

Instalar

Una vez descargado el instalador se ejecuta el archivo para comenzar la instalación del programa. Son seleccionados los componentes a instalar donde lo recomendable es instalar todos los paquetes que vienen por defecto y se indica el destino de la instalación del programa. Concretamente para Windows se elige la ruta por defecto que es [C:\xampp](#). Una vez realizados estos pasos se procede a finalizar la instalación del programa.

Probar

Se ejecuta XAMPP haciendo doble clic sobre el acceso directo del programa llamado *XAMPP Control Panel* donde muestra el panel de control del programa con los módulos instalados.

Arrancar los módulos Apache y MySQL pulsando los botones *Start* de cada uno, nos indican que están activados cuando aparece en el panel de control el *Running* al lado de cada uno de ellos. El panel de control se muestra en la imagen 1.4.

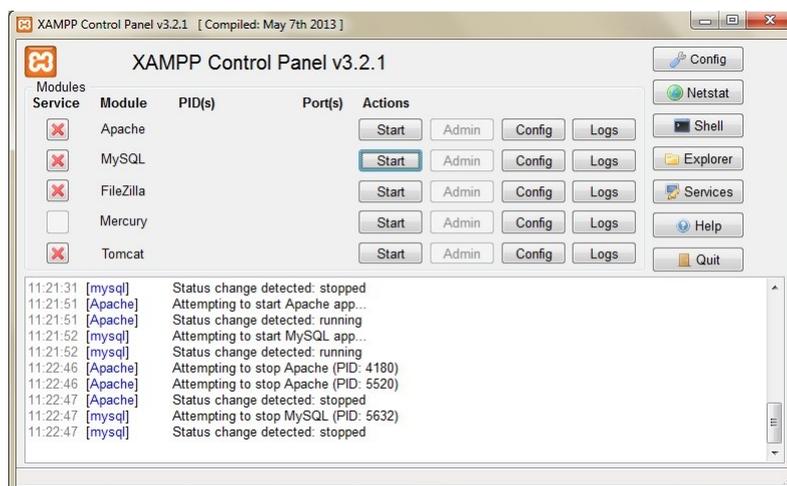


Figura 1.4 Panel de control de XAMPP

Pulsar el botón *Admin* de cada uno y aparecerá la página de inicio del servidor y la herramienta de administración de bases de datos *phpMyAdmin*. La

figura 1.5 muestra la página de administración de XAMPP donde se puede acceder a todas las funcionalidades de XAMPP.

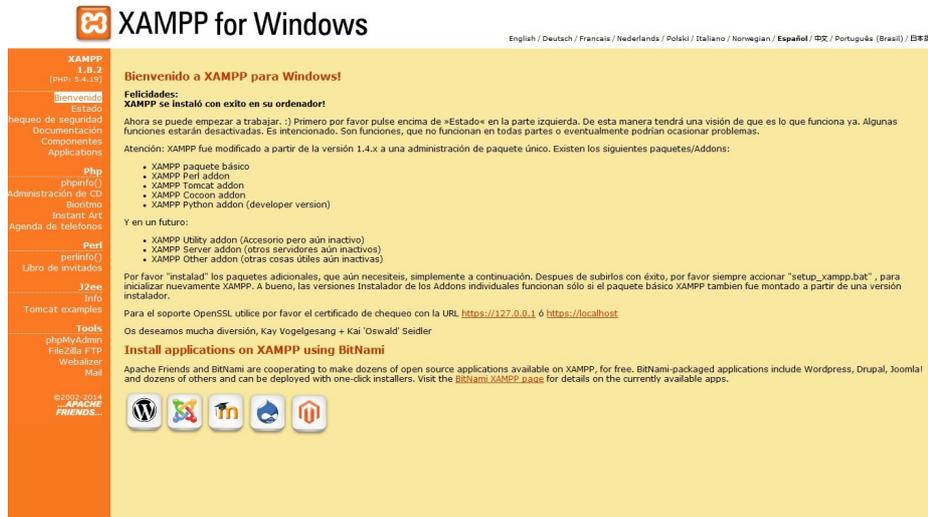


Figura 1.5 Página de administración de XAMPP

Directorio local

El directorio local es la ruta donde se guardan todos los proyectos creados. La ruta por defecto que ofrece XAMPP es `C:\xampp\htdocs`. Dentro de este directorio se crearán las carpetas que contendrán todos los proyectos que se utilizarán. Para WordPress se creará la carpeta `wordpress` y para Zend Framework 2 la carpeta `ZendF2`.

phpMyAdmin

Es una herramienta para la administración del servidor de bases de datos MySQL que se instala con XAMPP. Es de libre distribución y ofrece una interfaz gráfica que se maneja desde el navegador para realizar todo tipo de operaciones con las bases de datos: crear y borrar tablas, consultas y modificar datos, definir usuarios y dar permisos, etc. El panel de control de phpMyAdmin se muestra en la figura 1.6.

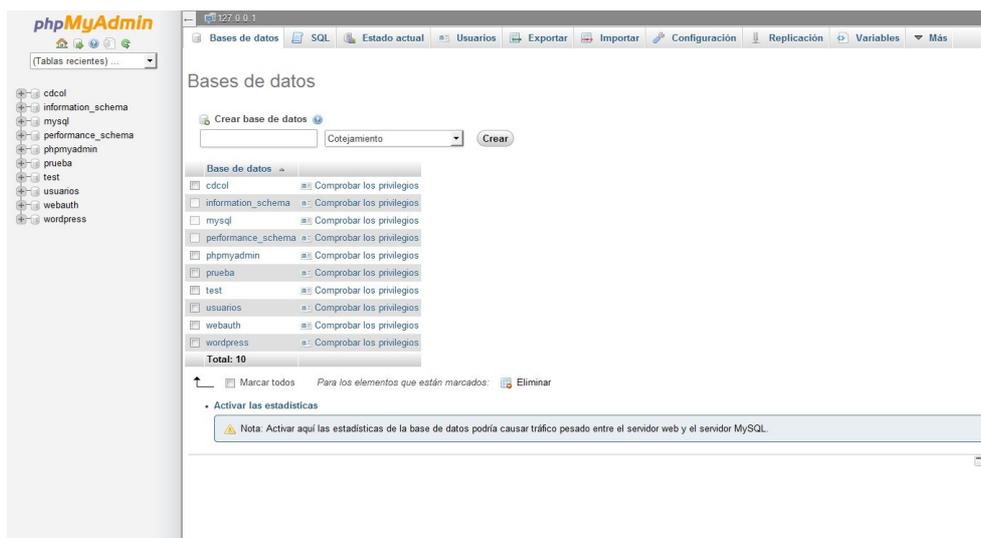


Figura 1.6 Página de administración de phpMyAdmin

Con phpMyAdmin se van a realizar todas las operaciones necesarias con base de datos en el servidor local para la creación de la base de datos en donde se va a instala WordPress y las operaciones con base de datos que realizaremos con Zend Framework 2.

1.3.2 Instalación de WordPress

WordPress es conocido por su facilidad de instalación. La mayor parte de las veces instalar WordPress es un proceso muy simple y rápido.

Con el servidor local ya disponible para su uso se procede a la instalación de WordPress, para ello, como primer paso se va a la página oficial de WordPress[10], portal web que se presenta en la Figura 1.7, y se descarga la última versión disponible del paquete de instalación. Al estar el paquete de WordPress en multitud de idiomas se accede a la versión española del portal web de WordPress y se descarga éste.

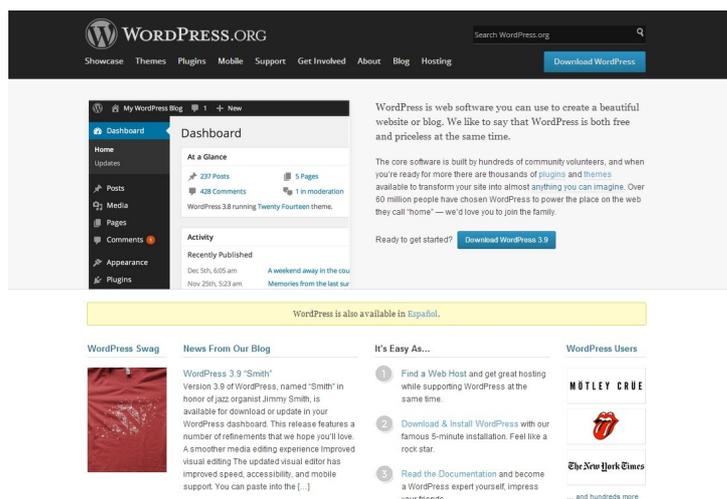


Figura 1.7 Portal de Wordpress

Como paso previo a la instalación hay que configurar el usuario y los permisos y crear la base de datos donde se sustentará el portal web localmente.

- Se ingresa en la página principal de phpMyAdmin y se configura el usuario 'root' para añadir una contraseña.
- Acceder al directorio donde se realiza la instalación de XAMPP y abrir con un editor de texto, por ejemplo el PHPStorm, el archivo `/xampp/phpmyadmin/config.inc.php` y se modifican los valores de los campos usuario a 'root'. En el campo contraseña se pondrá la misma que se puso en phpMyAdmin y el campo AllowNoPassword se modificará el valore a 'false'. En la figura 1.8 se muestra como queda una vez rellenados los campos mencionados.

```
18 /* Authentication type and info */
19 $cfg['Servers'][$i]['auth_type'] = 'config';
20 $cfg['Servers'][$i]['user'] = 'root';
21 $cfg['Servers'][$i]['password'] = '*****';
22 $cfg['Servers'][$i]['extension'] = 'mysql';
23 $cfg['Servers'][$i]['AllowNoPassword'] = false;
24 $cfg['Lang'] = '';
```

Figura 1.8 Campos a modificar para acceder a phpMyAdmin

- Se vuelve a acceder phpMyAdmin y se creará a continuación una base de datos de nombre 'wordpress' y tipo de cotejamiento 'utf8_unicode_ci' .

Una vez hechos estos pasos ya se está en disposición de descomprimir el

archivo descargado desde la web de WordPress dentro del directorio `/xampp/htdocs/`, quedando la instalación de WordPress en el directorio `/xampp/htdocs/wordpress`.

Completada la operación se accede al directorio y se edita el archivo `wp-config-sample.php` rellenando los campos `DB_NAME`, `DB_USER` y `DB_PASSWORD` con 'wordpress', 'root' y 'tu_contraseña' correspondiente como se ve en la figura 1.9. Entonces, se guarda el archivo con el nombre `wp-config.php`.

```
// ** Configuración de MySQL ** //
define('DB_NAME', 'wordpress'); // El nombre de la base de datos
define('DB_USER', 'username'); // Tu usuario de MySQL
define('DB_PASSWORD', 'password'); // ...tu contraseña
define('DB_HOST', 'localhost'); // 99% de la veces no necesitas cambiar este valor
```

Figura 1.9 Campos a modificar para acceder en wp-config.php

Ahora es el momento de abrir el navegador e introducir la siguiente dirección: `http://localhost/wordpress/`.

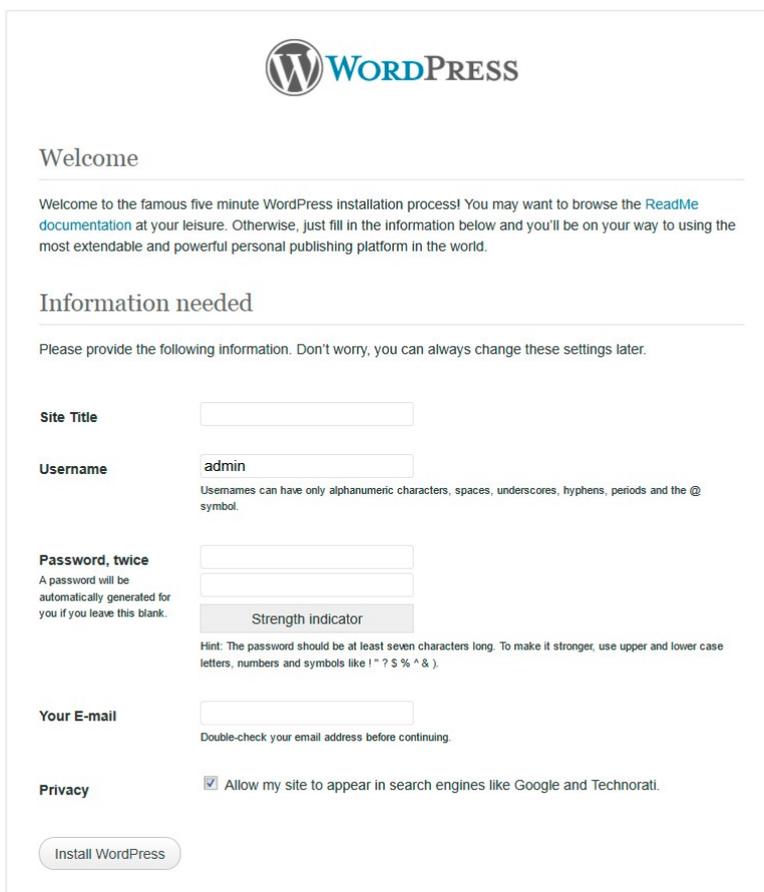


Figura 1.10 Formulario para terminar la instalación de WordPress

Con esta operación se accede directo a la pantalla de instalación de WordPress mostrada en la figura 1.10. Completando los datos requeridos se finaliza la instalación de WordPress. Si la instalación se ha hecho correctamente WordPress mostrará la pantalla indicando que todo el proceso ha sido exitoso como se muestra en la figura 1.11.



Figura 1.11 Pantalla que indica que la instalación exitosa de WordPress.

A partir de este momento para acceder a la instalación hecha en WordPress sólo es necesario introducir en el navegador la misma dirección mencionada anteriormente, *http://localhost/wordpress/*, autenticarse con el usuario y contraseña elegidos y ya se está en disposición para trabajar en el entorno WordPress. Una vez autenticados WordPress presenta al usuario la página de inicio como se muestra en la figura 1.12.

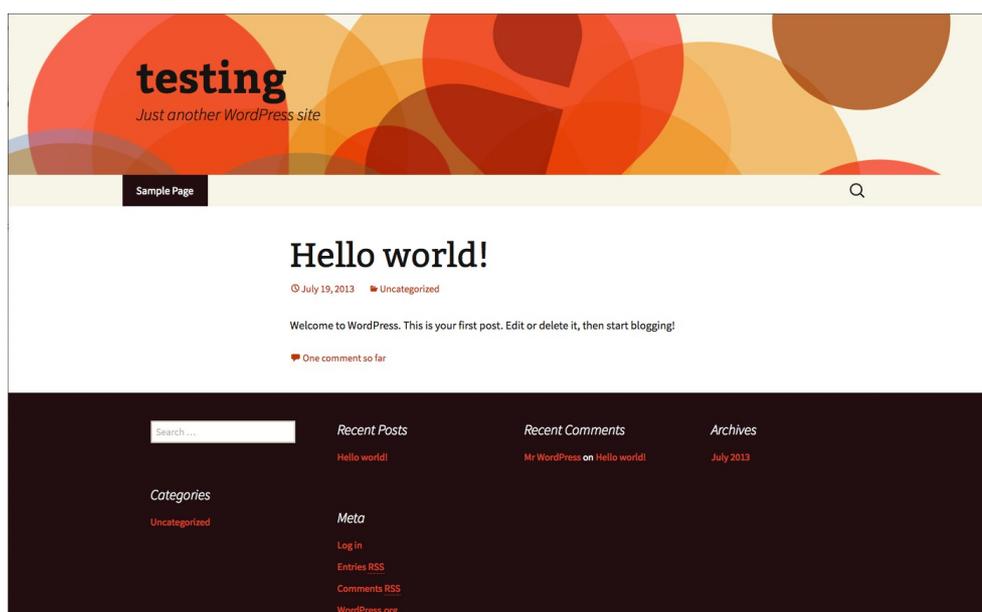


Figura 1.12 Pantalla al acceder por primera vez a WordPress

1.3.3 Instalación de ZF2

Para la instalación de Zend Framework 2 es necesario cumplir dos requisitos:

- ZF2 requiere un servidor web que tenga habilitada la extensión `mod_rewrite` o similar.
- PHP 5.3 o superior.

El programa XAMPP que se ha instalado en nuestro servidor local cumple con estos requisitos mínimos para la instalación. Otros elementos que hay que tener en cuenta antes de instalar ZF2 es tener instalado el programa Git y el Composer.

La instalación de Composer en Windows consiste en descargar de su página oficial[11] el instalador y ejecutarlo en nuestro ordenador.

Instalar ZF2 consiste en realizar las siguientes operaciones.

- Con la línea de comandos de Windows (CMD) hay que situarse dentro del directorio donde se desea realizar la instalación de ZF2. En el caso de ser el servidor local la ruta es `c://xampp/htdocs/`
- Se obtiene a través del comando `git` el `ZendSkeletonApplication` el cual brinda la estructura base para comenzar un nuevo proyecto en Zend Framework 2.

```
git clone git://github.com/zendframework/ZendSkeletonApplication.git
```

- Esta operación creará una carpeta llamada `ZendSkeletonApplication` el cual puede ser renombrada a conveniencia (para el ejemplo se llamará `zf2`). Si la operación se realizó con éxito nuestro directorio será como el mostrado a continuación en la figura 1.13.

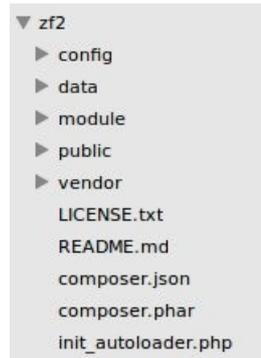


Figura 1.13 Directorio ZF2

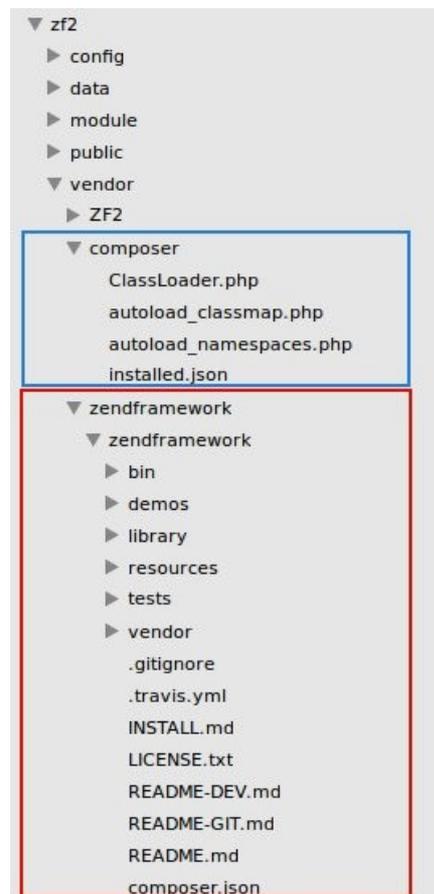


Figura 1.14 Estado del directorio al finalizar la instalación de Zf2

- Se accede dentro de la carpeta zf2 y se instalan las dependencias utilizando el Composer.

```
php composer.phar self-update (actualización del instalador)
php composer.phar install (instalación de la biblioteca ZF2)
```

- Una vez instaladas las dependencias, el directorio está compuesto por

nuevas carpetas correspondientes a la biblioteca de ZF2. El directorio final en el que se trabaja con ZF2 es como se muestra en la imagen 1.14 de la página anterior.

Realizados estos pasos, al ingresar en el navegador la dirección *http://localhost/zendf2/public/* aparece la siguiente página de bienvenida a Zend Framework 2 que muestra la figura 1.15.

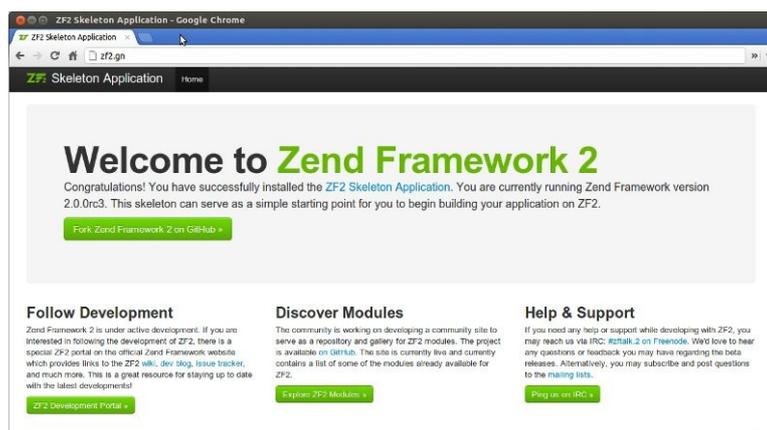


Figura 1.15 Pantalla inicial de Zf2

7.3.4 Instalación de WordPress en Heroku

Para la instalación y ejecución de WordPress en Heroku se han seguido varias guías de usuarios[12] que anteriormente se enfrentaron a este mismo problema ya que la documentación de Heroku no informa con precisión como realizar este procedimiento.

- Se realiza la clonación con la ayuda de Git de un repositorio desde Github y se le añade el nombre que se le va a dar a la aplicación.

```
$ git clone git://github.com/mhoofman/wordpress-heroku.git wp-appname
```

- Descarga desde la página oficial de Heroku e instalación de Heroku Gem, la herramienta de Heroku para conectar con la plataforma a través de Git para realizar tanto la instalación como cualquier modificación que se quiera hacer posteriormente.
- Una vez instalado se procede a crear la aplicación

```
$ cd wp-appname  
$ heroku create wp-appname
```

- Se crea la base de datos de PostgreSQL de la aplicación.

```
$ heroku addons:add heroku-postgresql:dev
```

Al realizar esta operación se obtiene la dirección url utilizada para acceder al servicio Heroku Postgres que tiene el siguiente formato

```
Attached as HEROKU_POSTGRESQL_COLOR por ejemplo HEROKU_POSTGRESQL_RED_URL
```

- Asignar la base de datos creada a su aplicación. Sustituyendo *COLOR* por el resultado entregado en la operación anterior.

```
$ heroku pg:promote HEROKU_POSTGRESQL_COLOR
```

- Crear una nueva rama de Github para cualquier cambio de configuración / instalación necesarios en el futuro.

```
$ git checkout -b production
```

- Modificar el archivo wp-config-sample.php a wp-config.php y actualizar las claves. Para ello WordPress ofrece un enlace donde copiar las claves a sustituir.

<https://api.wordpress.org/secret-key/1.1/salt/>

- Crear un commit inicial para guardar el final de la instalación.

```
$ git commit -m "Initial WordPress commit"
```

- Finaliza la instalación subiendo la aplicación a Heroku con la instrucción *push*.

```
git push heroku production:master
```

Terminada esta operación se accede a la web de Heroku y en el menú de usuario estará creada la aplicación web. La imagen 1.16 representa el menú de la web de Heroku en la que se ve la aplicación una vez instalada.

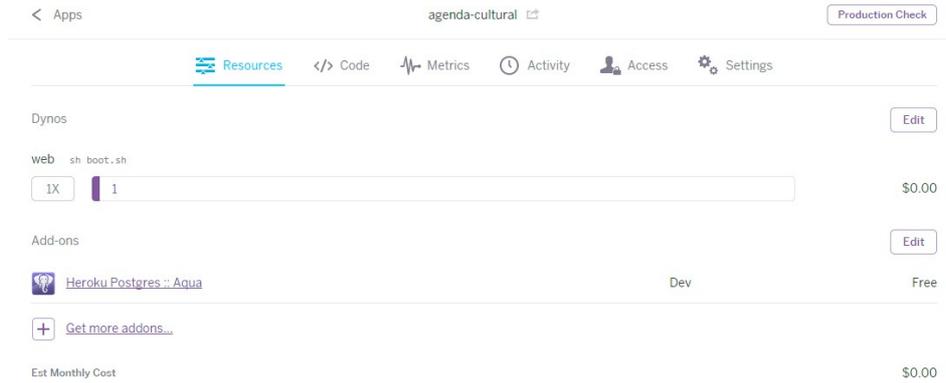


Figura 1.16 Menú de Heroku con la app instalada

Se accede a ella desde la url.

<http://nombre-de-tu-aplicacion.herokuapp.com/>

Para la creación de la aplicación web del proyecto se accede desde *<http://agenda-cultural.herokuapp.com>*

Pliego de Condiciones Legales

2.1 Concesión de licencia

Este programa es propiedad de la Universidad de Las Palmas de Gran Canaria y cualquier usuario debe estar de acuerdo y cumplir los términos y condiciones establecidas en esta licencia del programa, aceptando todas sus cláusulas. El uso de este software o de una copia en un PC, será bajo la autorización expresa del autor, tutor del proyecto y de la Escuela de Ingeniería de Telecomunicación y Electrónica de la Universidad de Las Palmas de Gran Canaria.

2.2 Derechos de autor

Este programa junto con la documentación están protegidos por las leyes de la propiedad intelectual que le sean aplicables, así como las disposiciones de los tratados internacionales. En consecuencia, el usuario debe utilizar el programa como cualquier producto protegido por derechos de autor. Sin embargo, el usuario podrá usar una copia y utilizar los códigos fuente de la programación y la documentación siempre bajo la autorización del autor, el tutor y de la Escuela de Ingeniería de Telecomunicación y Electrónica de la Universidad de Las Palmas de Gran Canaria.

2.3 Restricciones

El usuario no podrá realizar ingeniería inversa, de compilación o desensamblado del programa. Éste podrá transferir el programa a un tercero, siempre que no tenga copias del programa, incluyendo posibles actualizaciones o

retener material escrito adicional que acompañe al programa.

2.4 Limitación de responsabilidad

En ningún caso serán el autor ni el tutor, ni la Escuela de Ingeniería de Telecomunicación y Electrónica de la Universidad de Las Palmas de Gran Canaria responsables de los perjuicios directos, indirectos, incidentales o consiguientes, gastos, lucro cesante, pérdida de ahorros, interrupción de negocios, pérdida de información comercial o de negocio, o cualquier otra pérdida que resulte del uso o de la incapacidad de usar el programa o la documentación. El usuario conoce y acepta que los derechos de licencia reflejan esta asignación de riesgo como el resto de cláusulas y restricciones. El autor y los tutores rechazan cualquier otra garantía que no haya sido indicada anteriormente.

2.5 Varios

En el supuesto de que cualquier disposición de esta licencia sea declarada total o parcialmente inválida, la cláusula afectada será modificada convenientemente de manera que sea ejecutable una vez modificada, plenamente eficaz, permaneciendo el resto de este contrato en vigencia.

Este contrato se rige por las leyes de España. El usuario acepta la jurisdicción exclusiva de los tribunales de este país en relación con cualquier disputa que pudiera derivarse de la presente licencia.

Guías de usuario

3.1 Guía de usuario web

El usuario del portal web accede con la intención de conocer los eventos culturales que ofrecen los organismos oficiales de la isla de Gran Canaria.

La imagen 3.1 muestra la página de inicio del portal web. En lo más alto del portal web está el logotipo creado para identificar tanto en el portal web como en la aplicación móvil. A la derecha se muestran tres accesos directos a los perfiles del sistema en las redes sociales.

A continuación se accede al página de inicio donde se encuentra el menú principal con las secciones de la web: Inicio, Agenda, Calendario, Usuario, Contacto y Registrarse.



Figura 3.1 Página de inicio del portal web.

En la parte inferior y central se muestran los últimos eventos cultural añadidos al sistema indistintamente de la actividad y el lugar donde se realicen. Haciendo clic sobre alguno de ellos se procede a abrir la página correspondiente al evento.

A la derecha se presenta el *widget* de la web que permanece fijo en todas las secciones del portal web donde se muestran los eventos más próximos a realizarse. En él se informa del día en el que se va a llevar a cabo y un enlace que da la posibilidad al usuario de almacenarlo en su calendario propio de Google Calendar.

Accediendo en el menú principal en la sección “Agenda” se accede en el portal web a la página donde se muestra un menú que muestra todos los municipios de Gran Canaria. Esta acción se representa en la figura 3.2.

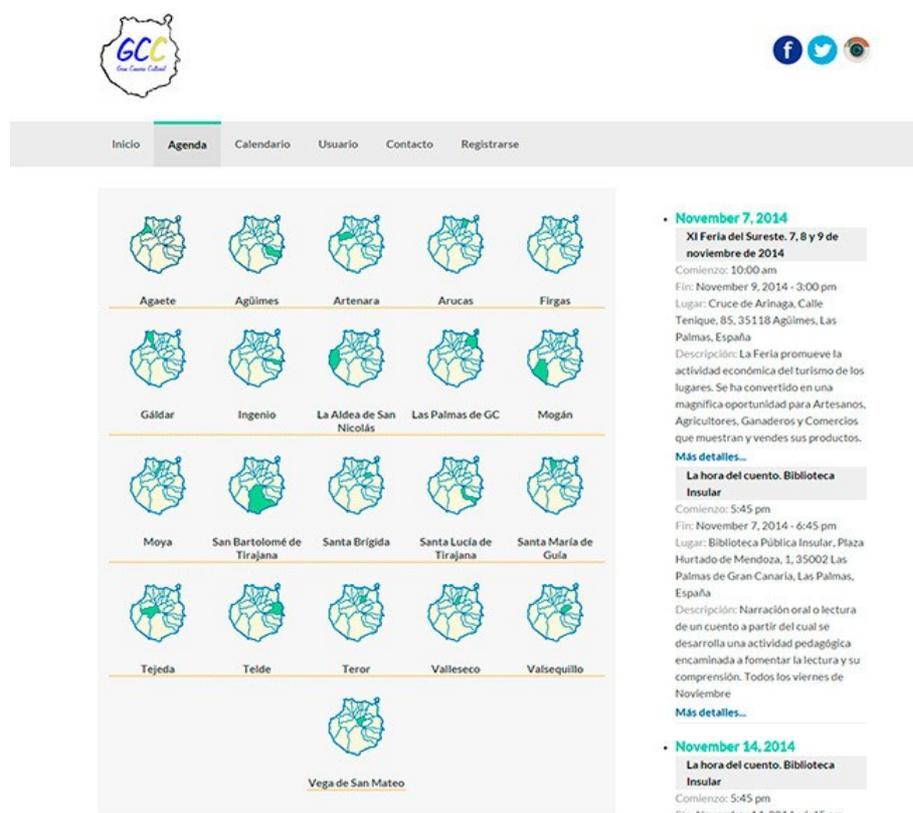


Figura 3.2 Vista de la sección Agenda.

Clicando sobre cada una de las imágenes se accede a un listado de los eventos ofertados por ese municipio. En la Figura 3.3 se muestra a modo de

ejemplo el resultado de clicar en el municipio de Agüimes, donde muestra el listado con los eventos, en este caso se muestra el único evento que hay para este municipio.



Figura 3.3 Lista de eventos de un municipio.

Para acceder a uno de estos eventos se hace clicando sobre el título y se muestra toda la información que se ofrece. En la Figura 3.4 se presenta como se ve un evento en el portal web.

Inicialmente se indica el título del evento y a continuación está toda la información que se le ofrece al usuario. Debajo del título está la información correspondiente a la fecha y hora a la que se realiza, información que se incluye rellenando los meta box creados para ello.

Seguidamente viene el cuerpo del evento que se compone de tres elementos que se pueden mostrar en el orden elegido por el editor, incluso puede ser que alguno de estos tres elementos no esté presente en las entrada de un evento.

- Texto que ofrece al usuario una información más detallada del evento en cuestión.
- Una imagen acorde a la temática del evento, siendo normalmente y siempre que exista el cartel creado para promocionar el evento.
- Recuadro con datos provenientes de la aplicación de Google Maps que

indica el lugar donde se realiza el evento.



Inicio Agenda Calendario Usuario Contacto Registrarse

XI Feria del Sureste. 7, 8 y 9 de noviembre de 2014

Subido el día 1 November 2, 2014 by ruymanrg

Este evento será el día **7 de noviembre de 2014**
Y comenzará a las **10:00**

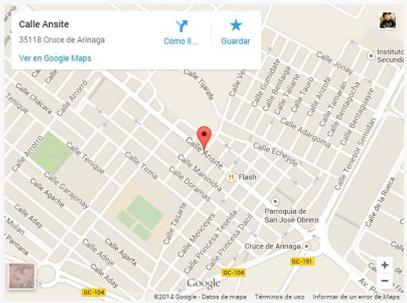


Los municipios de Agüimes, Ingenio y Santa Lucía, conscientes de que la mayoría de los turistas desconocen la riqueza cultural, tradicional y el propio entorno natural de la Isla que no corresponde a las zonas turísticas, decidieron desde el año 2004 realizar la Feria del Sureste, donde los participantes puedan conocer y disfrutar de los atractivos que ofrecen estos lugares a través de las tradiciones y costumbres de sus habitantes, la artesanía, la agricultura, la ganadería, el folklore y la alimentación (el cochino negro canario, la elaboración de aceite de oliva, la moda calada, los quesos y mojos que se exportan fuera de nuestras Islas, los tan premiados vinos, la apiterapia, etc., seguramente por separado o en su conjunto sorprenden al visitante foráneo y local).

La Feria, promueve la actividad económica del turismo de los lugares cuya orografía y comunicaciones no les brindan muchas oportunidades. Se ha convertido, tras 9 años de realización, en una magnífica oportunidad para Artesanos, Agricultores, Ganaderos y Comercios del Turismo Rural que muestran y venden sus productos en un bello ambiente favorecedor donde se mezclan sabores y ritmos musicales, y que se ubica estratégicamente cada año, de manera alterna, en lugares de fácil acceso y que son una de las principales vías comerciales de cada municipio: Avda. de Ansite, en el Cruce de Arinaga, T.M. Agüimes; calle Carlos V, en Carrizal, T.M. de Ingenio; y Avda. de Canarias, en Vecindario, T.M. de Santa Lucía.

Este año se realizará los días 7, 8 y 9 de noviembre en la avenida de Ansite, Cruce de Arinaga, en el municipio de Agüimes. Es posible gracias al entusiasmo, apoyo y participación de los agentes económicos y ciudadanía en general, así como por la valiosa colaboración de la Consejería de Empleo, Industria, Comercio y Artesanía del Cabildo de Gran Canaria.

Cada año tiene una duración de un fin de semana, para facilitar la participación de los expositores. En todas las ferias celebradas se ha pretendido una perfecta imbricación de la misma con las actividades comerciales de la zona, logrando que lejos de representar un problema sea un elemento de beneficio mutuo.



Este evento está postado en Eventos de Agüimes.

← La hora del cuento. Biblioteca Insular

Figura 3.4 Vista de un evento.

La imagen 3.5 muestra la sección del portal web es “Calendario”, aquí se presenta a los usuarios los eventos organizados por la fecha en el que se realizan

mediante la figura de un calendario donde se resalta de otro color el día que tiene eventos programados.

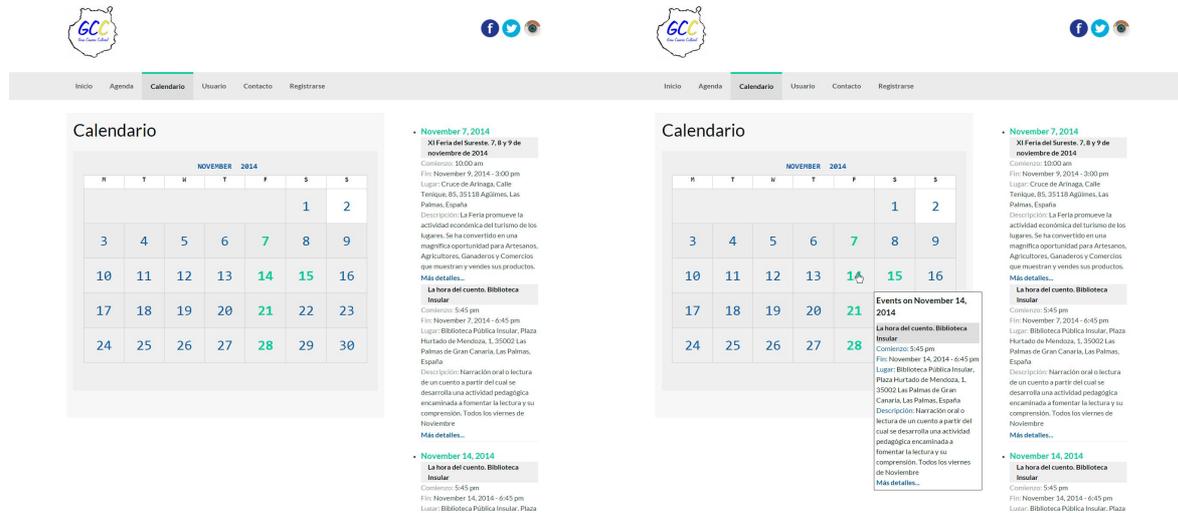


Figura 3.5 Sección Calendario.

Poniendo el ratón sobre los días remarcados por la presencia de eventos se muestra información sobre los eventos que se van a producir ese día como se puede comprobar en la segunda imagen de la Figura 3.5.

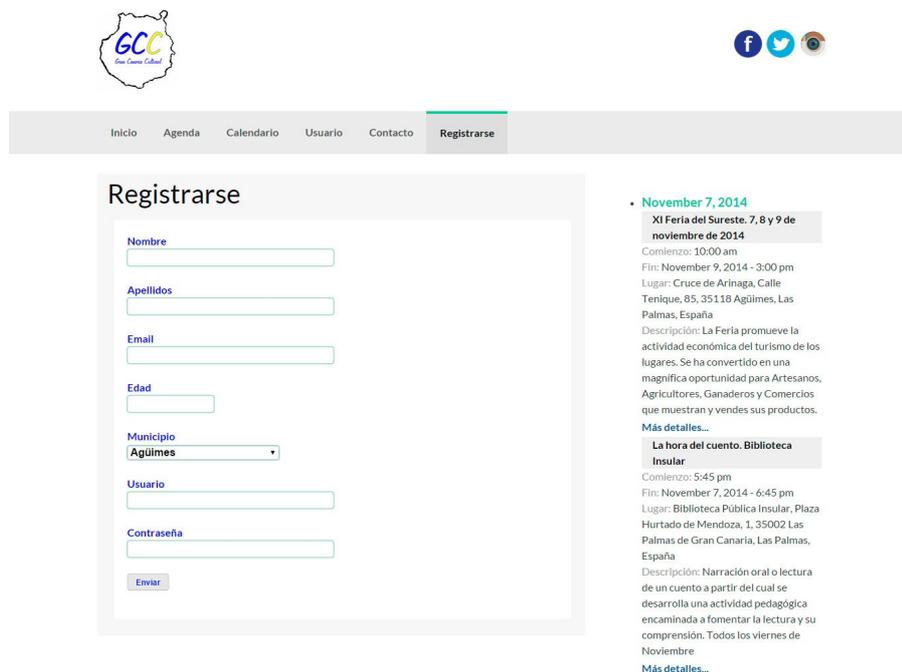


Figura 3.6 Sección Registrarse.

La figura 3.6 presenta otra de las secciones del portal web es la que se

muestra en el último lugar del menú, la sección “Registrarse”. Aquí al usuario se le presenta un formulario habitual en los portales en el que se piden datos personales del usuario, que rellenándolo y enviándolo se da de alta en el sistema.

La sección “Contacto” permite a los usuarios del portal web ponerse en contacto con la administración del sistema ante cualquier problema o sugerencia, esta acción se realiza mediante el envío de otro formulario que se recibe en la cuenta de correo de la administración. Esta acción se ve en la figura 3.7

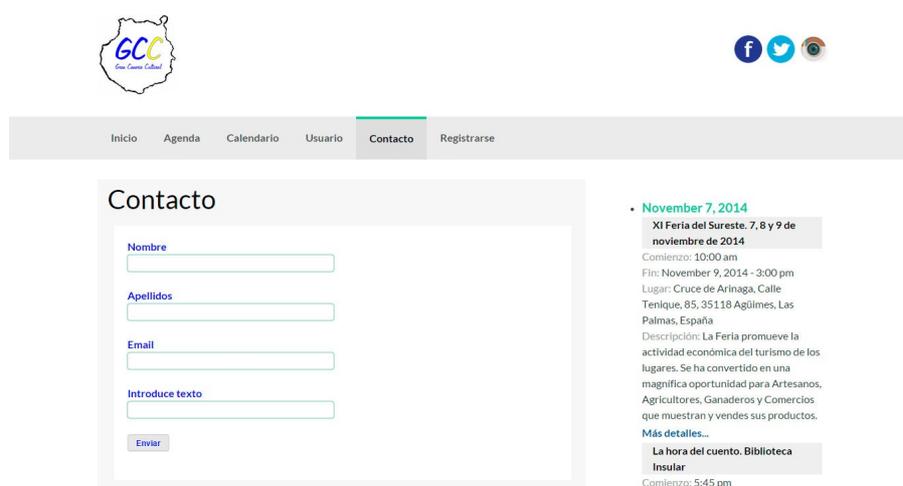


Figura 3.7 Sección Contacto.

La última sección del portal web es la correspondiente a “Usuario” que es mostrada en la imagen 3.8. Esta sección es para los usuarios que se han dado de alta previamente en el sistema. Introduciendo el nombre de usuario y contraseña establecidos en el momento de registrarse muestran los eventos programados en el municipio en el que el usuario se dio de alta. También da la posibilidad de darse de baja en el sistema.

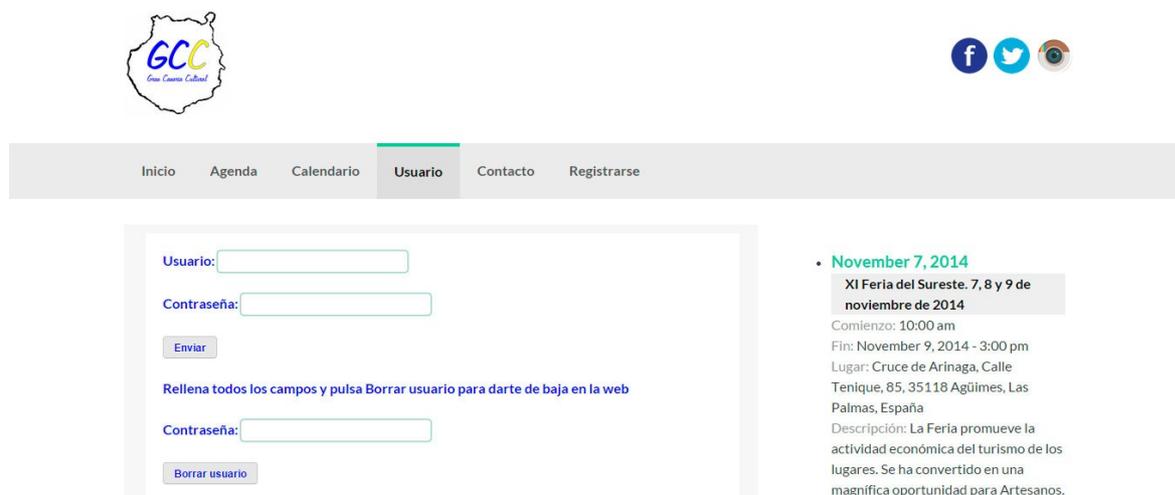


Figura 3.8 Sección Contacto.

3.2 Guía de usuario de la aplicación

Los usuarios de la aplicación disfrutarán de las mismas opciones tanto para la versión para Android como los aparatos que lo hacen bajo los. La interfaz de usuario sí que es diferente de un sistema operativo a otro. En las imágenes mostradas a continuación en la guía de usuario, las imágenes de la izquierda corresponden a la aplicación bajo el sistema Android y las de la derecha muestran la aplicación bajo el sistema operativo iOS.

Al iniciar la aplicación se muestran todos los eventos incluidos en el sistema, imágenes de la figura 3.9. La primera vez que se inicia la aplicación muestra todos los eventos sin filtrar, pero si se han aplicado alguno de los filtros disponibles, la aplicación almacena los filtros seleccionados y cada vez que se accede a la aplicación mostrará los eventos ofertados según los filtros que estén activos.

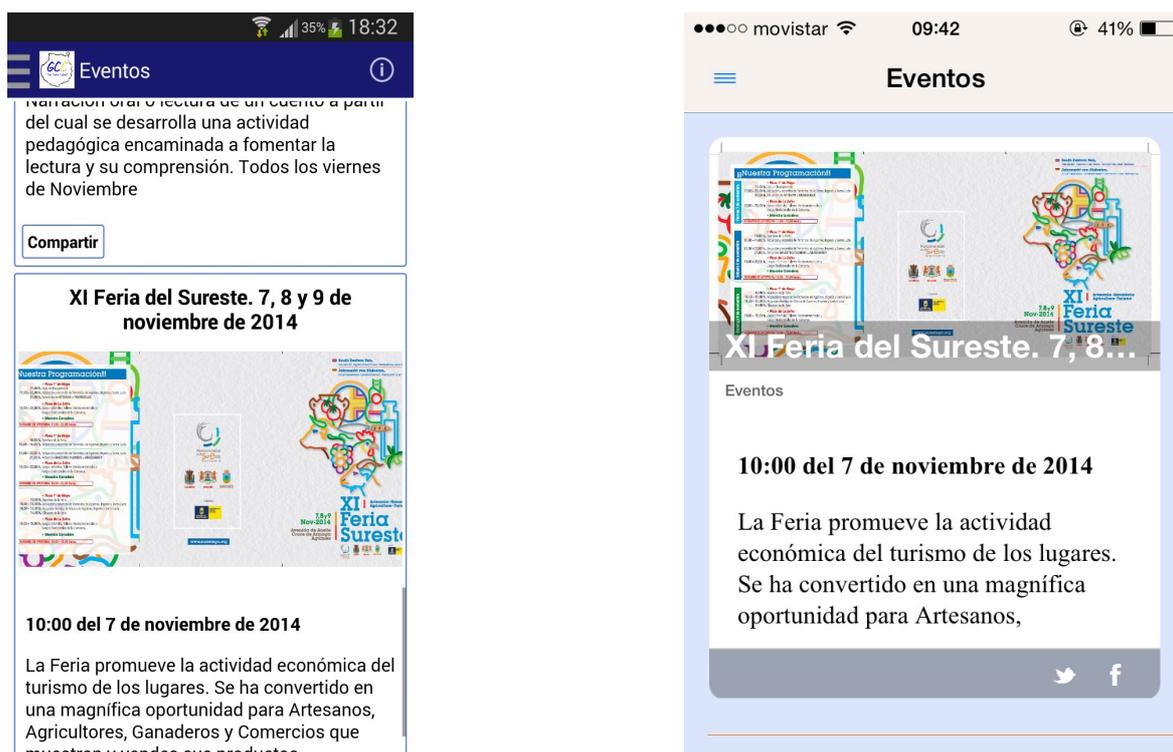


Figura 3.9 Pantalla de inicio de la aplicación.

En la parte de la izquierda de la pantalla se sitúa el menú de la aplicación, se utiliza para cambiar de sección dentro de la aplicación y para aplicar los distintos filtros que ofrece la aplicación.

En la imagen 3.10 se observa como inicialmente se muestra el logo de la aplicación seguido del menú principal que las secciones Eventos y Redes Sociales. A continuación se muestran los filtros de los eventos tanto por municipios como por actividad.

Si se accede a la sección eventos se muestra los eventos incluidos en el sistema como se muestra en la imagen 3.9, si por el contrario se accede a la sección Redes Sociales la aplicación presenta en pantalla el menú con las redes sociales a las que conectarse. Esta sección de la aplicación se muestra en la figura 3.11 de la siguiente página.

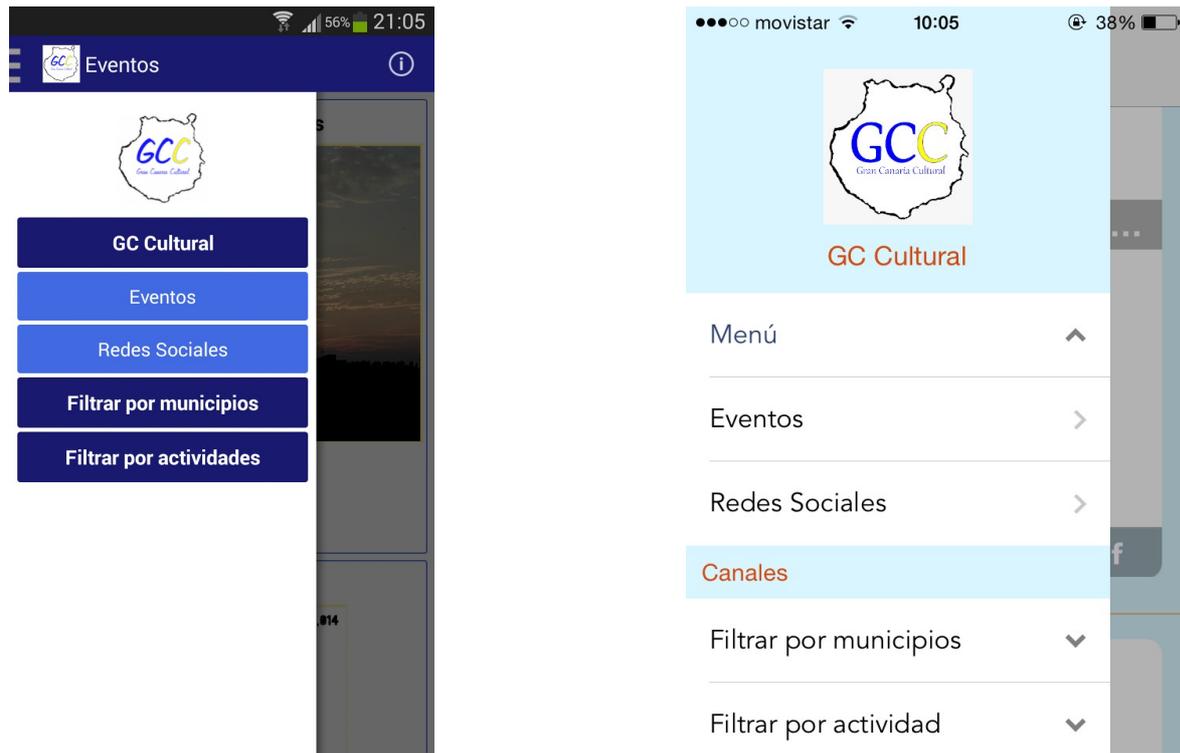
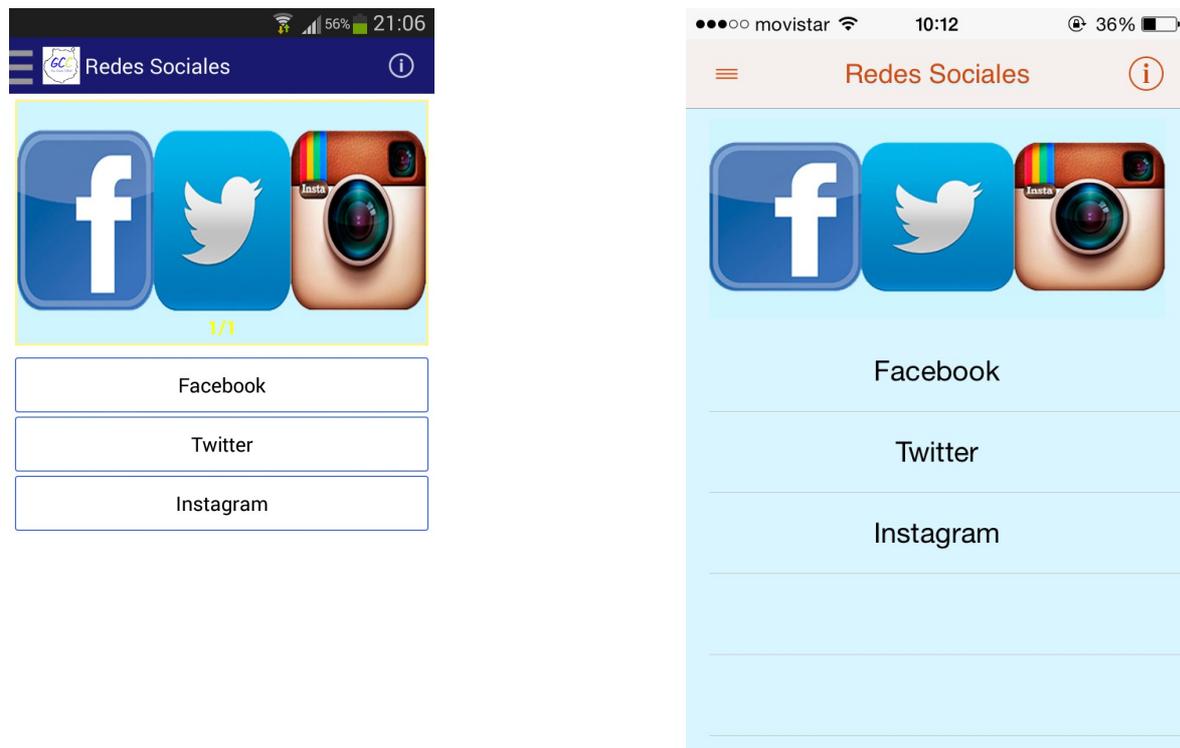


Figura 3.10 Menú de la aplicación.



3.11 Sección Redes Sociales.

Accediendo sobre cualquiera de las opciones mostradas, la aplicación

redirecciona a la url del perfil de usuario del sistema de la red social elegida. En la figura 3.12 se muestra cómo se comporta la aplicación al acceder al perfil de Facebook del sistema. Figura

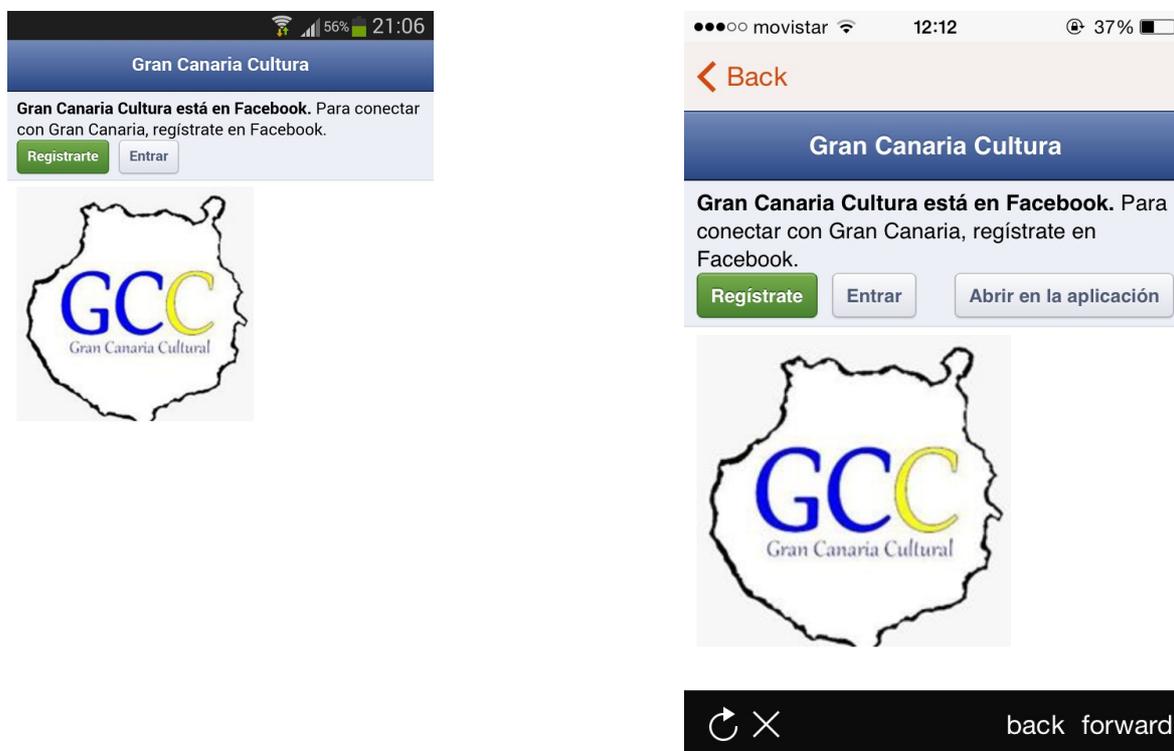


Figura 3.12 Redirección de la aplicación a una red social.

Las siguientes opciones del slide menú son los filtros que actúan sobre los eventos para limitar el número de eventos a mostrar para que los usuarios tengan un acceso más rápido al tipo de eventos que son de su interés. Para activar estos filtros se accede al slide menú y se despliega el filtro que se quiera activar o desactivar y se marca el filtro deseado. Como ejemplo, en la imagen 3.13 muestra como se accede al apartado de los filtros por municipios y es activado el filtro *Agüimes*, una vez activado este filtro se muestran en la aplicación los eventos de la categoría *Agüimes*.

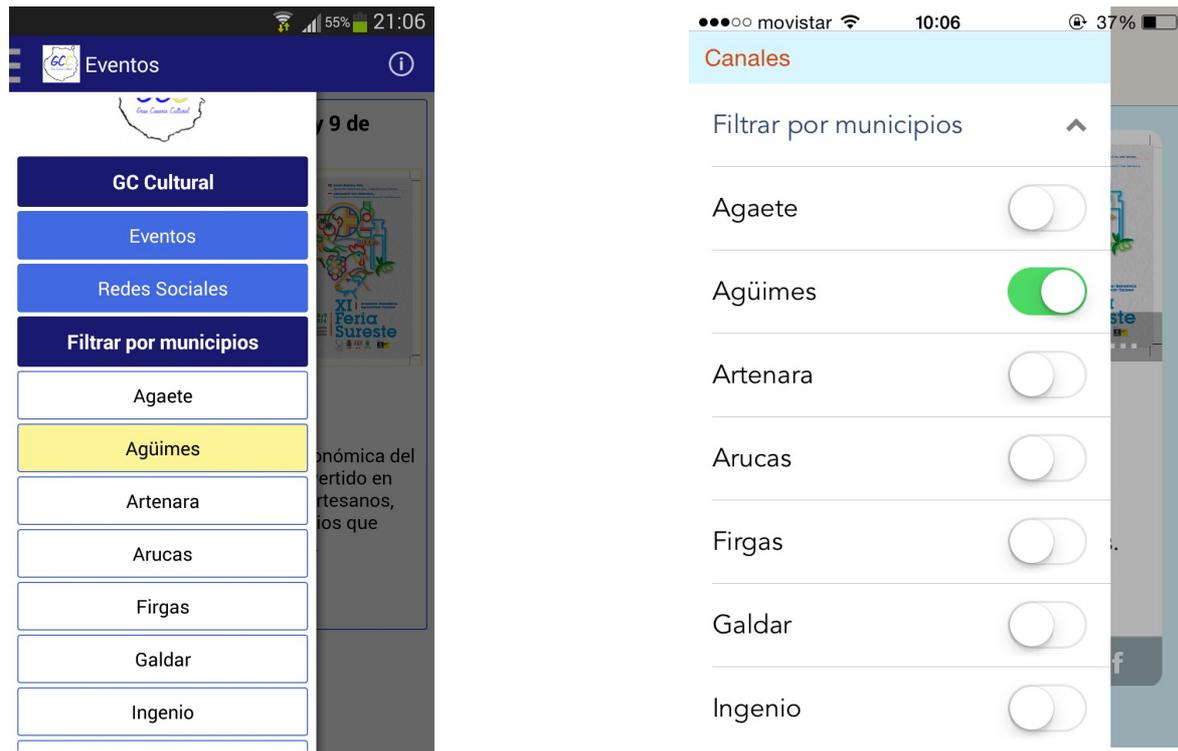


Figura 3.13 Activación de un filtro.

La aplicación da la opción al usuario de activar más de un filtro simultáneamente, si ésto ocurre se muestra en pantalla todos los eventos que contengan alguna de las categorías activadas a través de los filtros.

Si el usuario desactiva alguno de los filtros la aplicación vuelve al estado anterior donde muestra los eventos según los filtros que queden activos o en el caso que no haya ningún filtro activo muestra la totalidad de los eventos incluidos en el sistema.

La imagen 3.14 muestra el último apartado del menú principal,son dos secciones más que contiene la aplicación. El primero es el que se muestra en la figura 3.15, la sección enviar email, que ofrece la posibilidad al usuario de contactar con la administración del sistema mediante el envío de un correo electrónico.

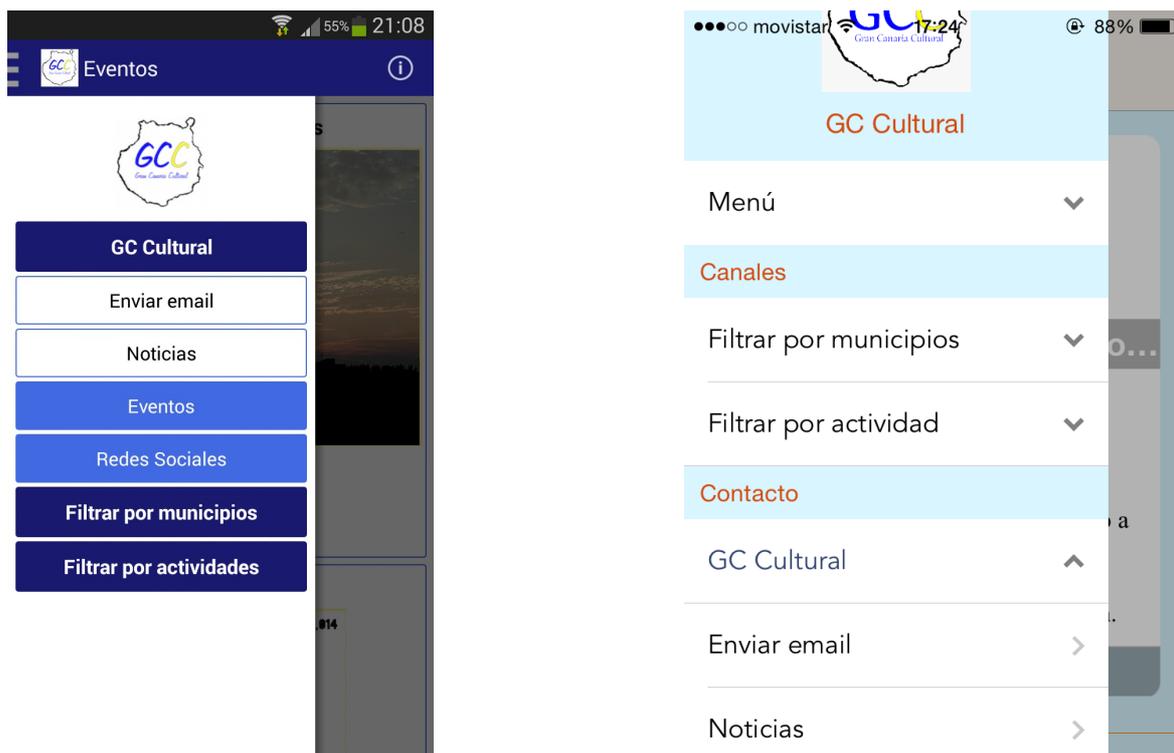


Figura 3.14 Menú de la aplicación.

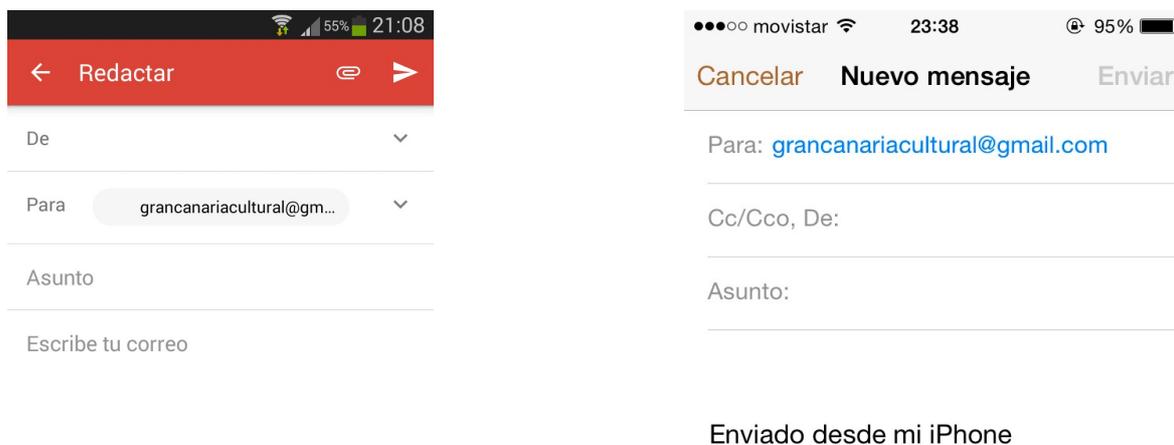


Figura 3.15 Pantalla para enviar email al sistema.

La segunda sección, la representada en las imágenes de la figura 3.16, da

la posibilidad de mostrar todos los eventos del sistema en formato RSS para que el usuario tenga toda la información que ofrece el sistema actualizada de forma automática y la pueda compartir mediante blogs u otros portales web.



Figura 3.16 Sistema RSS de la aplicación.

Presupuesto

Presupuesto

El presupuesto que se desglosa a continuación es el correspondiente al Proyecto Fin de Carrera que implica el trabajo desde el planteamiento inicial para el estudio de las herramientas a utilizar, la creación de código para el portal web, la configuración de los servicios en la nube, etc., hasta la redacción de la memoria. El presupuesto se divide en los siguientes apartados:

- Tiempo empleado en la realización del PFC.
- Amortización de las herramientas de Hardware y Software.
- Presupuesto final.

1.1 Tiempo empleado en la realización del PFC

Según el Colegio Oficial de Ingenieros Técnicos de Telecomunicación (COIT), se han eliminado los baremos orientativos por el Ministerio de Economía y Hacienda, siguiendo las directivas europeas. Esto indica que los honorarios de mano de obra de un ingeniero son libres.

En este proyecto se va a establecer que el coste por hora de trabajo es de 65 € para horario normal y 70 € fuera de la jornada normal. En base a estos precios se emplea el siguiente cálculo para obtener los honorarios.

$$H = (H_n * 65) + (H_e * 70)$$

Donde:

H = Honorario total

H_n = Horas en jornada normal

H_e = Horas fuera de la jornada normal

Además hay que aplicar un coeficiente de reducción a los honorarios calculados en función de las horas dedicadas a la realización del proyecto. Los coeficientes que se van a aplicar son los expuestos en la tabla 1.1.

Coeficiente de reducción	
Número de horas	Coeficiente
Hasta 36 horas	1
36-72 horas	0,9
72-108 horas	0,8
108-144 horas	0,7
144-180 horas	0,65
180-360 horas	0,6
360-510 horas	0,55
510-720 horas	0,5
720-1080 horas	0,45
Más de 1080 horas	0,4

Tabla 1.1 Coeficiente de reducción

Teniendo en cuenta que la duración del proyecto ha sido de 10 meses, 40 semanas, y se ha trabajado 7 horas diarias durante 5 días a la semana, el número total de horas trabajadas son:

$H_n = 40 \text{ semanas} \times 5 \text{ días a la semana} \times 7 \text{ horas al día} = 1400 \text{ horas}$ trabajadas en horario de jornada normal.

El cómputo de horas trabajadas en horas fuera de la jornada normal es cero, por lo que tenemos $H_e = 0$.

Aplicando el coeficiente de reducción de la tabla 1.1 se obtiene que se debe aplicar un coeficiente de 0,4 debido a que el número de horas es superior a 1080 horas.

Aplicando todos los valores obtenidos finalmente el coste total de recursos humanos queda así:

$$H = (1400 \times 65 + 0 \times 70) \times 0,4 = 36.400 \text{ €}$$

Por lo que los honorarios por mano de obra de ingeniero en el proyecto asciende a **treinta y seis mil cuatrocientos euros**.

1.2 Amortización de las herramientas de Hardware y Software

En la amortización de inmovilizados se tienen en cuenta todos los equipos o elementos necesarios para la elaboración del proyecto. Cada uno de estos elementos tiene un coste y un periodo de vida útil, generalmente superior al tiempo empleado en una tarea concreta. Esta amortización, que se muestra en las siguientes tablas, se calcula con la siguiente expresión:

$$CA=(CAD-VR)/VU$$

Donde:

CA = Cuota anual.

CAD= Cuota de adquisición. Valor invertido en la adquisición de la herramienta.

VR= Valor residual. Valor que se estima que tendrá la herramienta al final de su vida útil. El valor típico fijado es de un 5 % de su valor de adquisición.

VU= Años de vida útil. Años que se estima que la herramienta conserva un provecho pleno y competitivo.

Para el hardware y el software se supone un tiempo de amortización de 3 años.

- **Herramientas de hardware**

Las herramienta hardware necesarios para el desarrollo de este proyecto se incluye un ordenador personal durante un periodo de 10 meses para la realización y redacción del proyecto y dos móviles para la comprobación de la aplicación. Las características del ordenador y los móviles vienen especificadas en la tabla 1.2. Además, en la tabla 1.2 se muestra el cálculo para obtener la amortización de las herramientas de hardware.

Descripción	CAD (€)	VR (€)	VU (años)	CA (€)	TU (años)	AM (€)
Ordenador Personal Intel Core i7-860 2.8Ghz	800 €	40 €	3 años	253,33 €	10/12 años	211,11 €
Samsung GALAXY S III	500 €	25 €	3 años	158,33 €	2/12 años	26,39 €
Iphone 5S	430 €	21,5 €	3 años	136,17 €	2/12 años	22,69 €
Coste total de recursos hardware						260,19 €

Tabla 1.2 Coste total de herramientas de hardware

- **Herramientas de software**

Se describe el software utilizado para el desarrollo del proyecto que es necesaria una licencia para su uso. Para la realización del proyecto únicamente ha sido necesario un software con licencia, el sistema operativo Microsoft Windows 7. El resto del software utilizado tiene licencia gratuita. El cálculo para obtener la amortización de las herramientas de software viene dado en la tabla 1.3.

Descripción	CAD (€)	VR (€)	VU (años)	CA (€)	TU (años)	AM (€)
Microsoft windows 7	99 €	4,95 €	3 años	31,35 €	10/12 años	26,13 €
Coste total de recursos software						26,13 €

Tabla 1.3 Coste total de herramientas de software

1.3 Presupuesto total del Proyecto Fin de Carrera

El presupuesto total del Proyecto Fin de Carrera se obtiene mediante la suma de los resultados de los apartados anteriores. En la tabla 1.4 se especifica cada uno de los apartados calculados y su precio y la suma total.

Presupuesto total sin impuestos	
Concepto	Coste
Recursos humanos	36.400 €
Inmovilizado material (hardware)	260,19 €
Inmovilizado material (software)	26,13 €
Total	36.686,32 €

Tabla 1.4 Presupuesto total sin impuestos

El presupuesto final sin impuestos asciende a **treinta y seis mil seiscientos ochenta y seis euros y treinta y dos céntimos**

A este presupuesto se le debe aplicar un cargo por Impuestos de la Comunidad Autónoma de Canarias (I.G.I.C), que actualmente asciende a un 7%. De esta forma el presupuesto final queda como sigue en la siguiente tabla:

Presupuesto total	
Concepto	Coste
Presupuesto sin impuestos	36.686,32 €
IGIC 7 %	2568,04 €
Total	39254,36 €

Tabla 1.5 Presupuesto total

El presupuesto total asciende a **treinta y nueve mil doscientos cincuenta y cuatro euros y treinta y seis céntimos.**

Fdo.:Ruymán Rodríguez González