

Trabajo de Fin de Grado

Grado en Ingeniería Informática

Sistema de Ubicación Geográfica de RSS a través de PLN

RSS Geographical Location System via PLN

Moisés Yanes Carballo

La Laguna, 21 de febrero de 2017

D. **José Luis González Ávila**, con N.I.F. 78.677.390-W profesor Asociado de Universidad adscrito al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como tutor

C E R T I F I C A (N)

Que la presente memoria titulada:

“Sistema de Ubicación Geográfica de RSS a través de PLN”

ha sido realizada bajo su dirección por D. **Moisés Yanes Carballo**, con N.I.F. 78.644.323-D.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 25 de Agosto de 2017.

Agradecimientos

Con la realización y finalización de este proyecto concluye la que ha sido una etapa larga y que ha traído consigo momentos inolvidables. Aunque en este largo periodo de tiempo ha habido situaciones más delicadas y difíciles siempre he trabajado para mejorar y superarlas.

Nada de esto hubiera sido posible sin mis compañeros, ellos han sido una pieza fundamental para llegar hasta aquí y es a ellos a quien les debo una pequeña parte de este éxito.

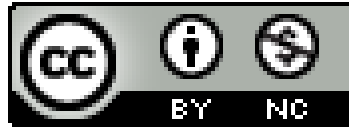
Sin duda alguna he de nombrar a los profesores que me han impartido clase durante estos años, los conocimientos impartidos me han permitido crecer como informático, además, la buena actitud y predisposición mostrada por ellos ha ayudado aún más si cabe.

Mención especial para mi tutor de TFG que me ha prestado su ayuda en todo momento, buscando alternativas para vernos y aclarar todas las dudas sobre el desarrollo del proyecto que me pudieran surgir.

Por último y no menos importante, agradecer a mi familia el apoyo que me han dado durante estos años y comprenderme a pesar de pasar horas y horas delante del ordenador.

Muchas gracias a todos.

Licencia



© Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial 4.0 Internacional.

Resumen

El objetivo de este trabajo ha sido la creación de una aplicación web que permita clasificar noticias de la zona de Santa Cruz de Tenerife y en menor medida de San Cristóbal de La Laguna, así como ubicarlas en el lugar exacto en donde se han producido cada una de ellas.

De esta forma se podrá clasificar los problemas que afectan a una ubicación en concreto. Entre la información de la noticia que podremos consultar están: título, descripción, categoría y link a la noticia.

Además de esto la aplicación permite realizar una búsqueda por ubicación y descargar las noticias para una localización concreta en diferentes formatos.

Para realizar este trabajo se han puesto en práctica conocimientos en gestión de sistemas, desarrollo web, gestión de base de datos y, diferentes lenguajes de programación, así como técnicas de Procesamiento de Lenguaje Natural.

En lo referente a la base de datos se trata de una base de datos no relacional, lo cual, en lugar de guardar los datos en tablas como se hace en las bases de datos relacionales, se guarda en estructuras de datos en documentos similares a JSON con un esquema dinámico. En concreto, MongoDB y consta de dos colecciones

Por último, para el diseño del servicio web se ha usado el framework Bootstrap y para la creación del mapa se ha hecho uso de la Google Maps JavaScript API.

Palabras clave: Aplicación web, noticias, ubicación, Santa Cruz de Tenerife, mapa

Abstract

The objective of this work has been the creation of a web application that allows to classify news of the zone of Santa Cruz de Tenerife and to a lesser extent of San Cristóbal de La Laguna, as well as to locate them in the exact place where they have occurred each one of them.

This way you can classify the problems that affect a location. Among the information of the news that we can consult are: title, description, category and link to the news.

In addition to this, the application allows to perform a search by location and to download the news for a specific location in different formats.

To carry out this work, have been put into practice knowledge in systems management, web development, database management and, different programming languages, as well as techniques of Natural Language Processing.

Regarding the database it is a non-relational database, which, instead of storing the data in tables as it is done in relational databases, is stored in data structures in documents similar to JSON with a dynamic scheme. In particular, MongoDB and consists of two collections

Finally, for the design of the web service, has been used the Bootstrap framework and for the creation of the map has been made use of Google Maps JavaScript API

Keywords: *Keywords: Web application, news, location, Santa Cruz de Tenerife, map*

Índice General

Capítulo 1. Introducción	5
1.1 Big Data.....	5
1.1.1 Capas Tecnológicas	5
1.1.2 Características.....	7
Capítulo 2. Historia del BigData	8
2.1 Antecedentes	8
2.2 Actualidad.....	8
Capítulo 3. Planificación y objetivos del proyecto	9
3.1 Gestión de RSS	9
3.2 Creación de la base de datos	9
3.3 Generación del mapa.....	10
3.4 Diseño de la aplicación web.....	11
Capítulo 4. Herramientas utilizadas	12
4.1 Tratamiento de RSS.....	12
4.1.1 RSS.....	12
4.1.2 Java	13
4.2 Base de datos	13
4.2.1 MongoDB.....	13
4.2.2 PHP	14
4.3 Alojamiento Web	14
4.3.1 Apache.....	14
4.3.2 IAAS ULL.....	14
4.4 Diseño Web.....	15
4.4.1 Bootstrap.....	15
4.4.2 HTML5, CSS3, JavaScript, JQuery	15
4.4.3 JSON	16
4.4.4 FPDF.....	16
4.5 Mapa.....	16

4.5.1	Google Maps JavaScript API.....	16
Capítulo 5.	Aplicación Web	17
5.1	Arquitectura de la aplicación.....	17
5.2	Estructura de directorios.....	18
5.3	Implementación.....	19
5.3.1	Gestión de RSS.....	19
5.3.2	Creación y gestión de la base de datos.....	22
5.3.3	Generación del mapa.....	28
5.3.4	Diseño de la aplicación web.....	29
5.4	Resultados.....	31
5.4.1	Resoluciones.....	31
5.4.2	Presentación de resultados.....	33
Capítulo 6.	Problemas encontrados	36
6.1.1	Eliminación de noticias de la base de datos.....	36
6.1.2	Generación de mapa.....	37
6.1.3	Superposición de marcadores.....	37
6.1.4	Movimiento horizontal en divs.....	39
Capítulo 7.	Conclusiones y líneas futuras	40
Capítulo 8.	Summary and Conclusions	42
Capítulo 9.	Presupuesto	44
9.1	Estimación coste de herramientas.....	44
9.2	Estimación coste de desarrollo e implantación de la aplicación.....	45
9.3	Presupuesto final estimado.....	45
Apéndice A.	Diagrama de prueba	46
A.1.	Graficas de análisis.....	49
Bibliografía		51

Índice de figuras

Figura 1.1. Capas tecnológicas.....	6
Figura 1.2. Conceptos relacionados con BigData.....	7
Figura 4.1. RSS.....	12
Figura 4.2. MongoDB y PHP.....	14
Figura 4.3. Apache.....	15
Figura 4.4. JSON.....	16
Figura 5.1. Arquitectura aplicación.....	17
Figura 5.2. Estructura de directorios de la aplicación.	18
Figura 5.3. Constructor cliente MongoDB.....	19
Figura 5.4. Constructor de rss.....	20
Figura 5.5. Vector de rss.....	20
Figura 5.6. Constructor sobrecargado de noticias.....	21
Figura 5.7. Comparaciones para el cálculo de la ubicación.....	25
Figura 5.8. Actualización de la ubicación de las noticias.....	26
Figura 5.9. Inserción de coordenadas por medio de csv	27
Figura 5.10. Inicialización de mapa.....	28
Figura 5.11. Ampliar mapa Figura 5.12. Reducir mapa.....	30
Figura 5.13. Resolución 1920x1080.....	32
Figura 5.14. Resolución 1366x600	32
Figura 5.15. Vista inicial.....	33
Figura 5.16. Vista de marcadores y noticias.....	33
Figura 5.17. Visualización de información de noticias	34
Figura 5.18. Generación de PDF.....	35
Figura 6.1. Agrupación de marcadores en clústeres.....	38

Índice de tablas

Tabla 9.1. Estimación de coste de herramientas.....	44
Tabla 9.2. Estimación de coste de desarrollo.....	45

Capítulo 1.

Introducción

En este apartado se explica de forma resumida que es el Big Data y cuáles son sus características principales

1.1 Big Data

Este término hace referencia a grandes volúmenes de datos (estructurados, semiestructurados y no estructurados) que no pueden ser procesados por sistemas tradicionales y a partir de los cuales mediante un proceso de transformación se puede obtener información útil.

En lo referente a nuestro proyecto, es necesario el empleo de técnicas de Big Data puesto que se precisa del análisis de grandes cantidades de datos semiestructurados como son los RSS de noticias. Aplicando estas técnicas se pretende poder disponer de una aplicación capaz de manipular dichos datos y mediante diferentes métodos y técnicas de PLN obtener información útil. En nuestro caso, transformaremos datos simples semiestructurados en noticias ubicadas geográficamente dentro de un mapa, además de diferentes estadísticas que se pueden extraer de los propios datos.

1.1.1 Capas Tecnológicas

Todo proyecto centrado en Big Data se centra en al menos las siguientes capas tecnológicas.

1.1.1.1 Almacenamiento

Los recursos hardware y software permiten el almacenamiento distribuido y redundante de los datos, facilitando su acceso y disponibilidad, y evitando los silos de datos.

I.1.1.1.2 Procesamiento

Las herramientas de procesamiento de los datos dan la base tecnológica para operar con grandes volúmenes de información en tiempo real.

I.1.1.1.3 Análisis

Los algoritmos y métodos realizan el análisis de los datos, produciendo información de valor.

Las capas tecnológicas y la distribución de las mismas dentro de la aplicación desarrollada es la que se muestra en la imagen siguiente:

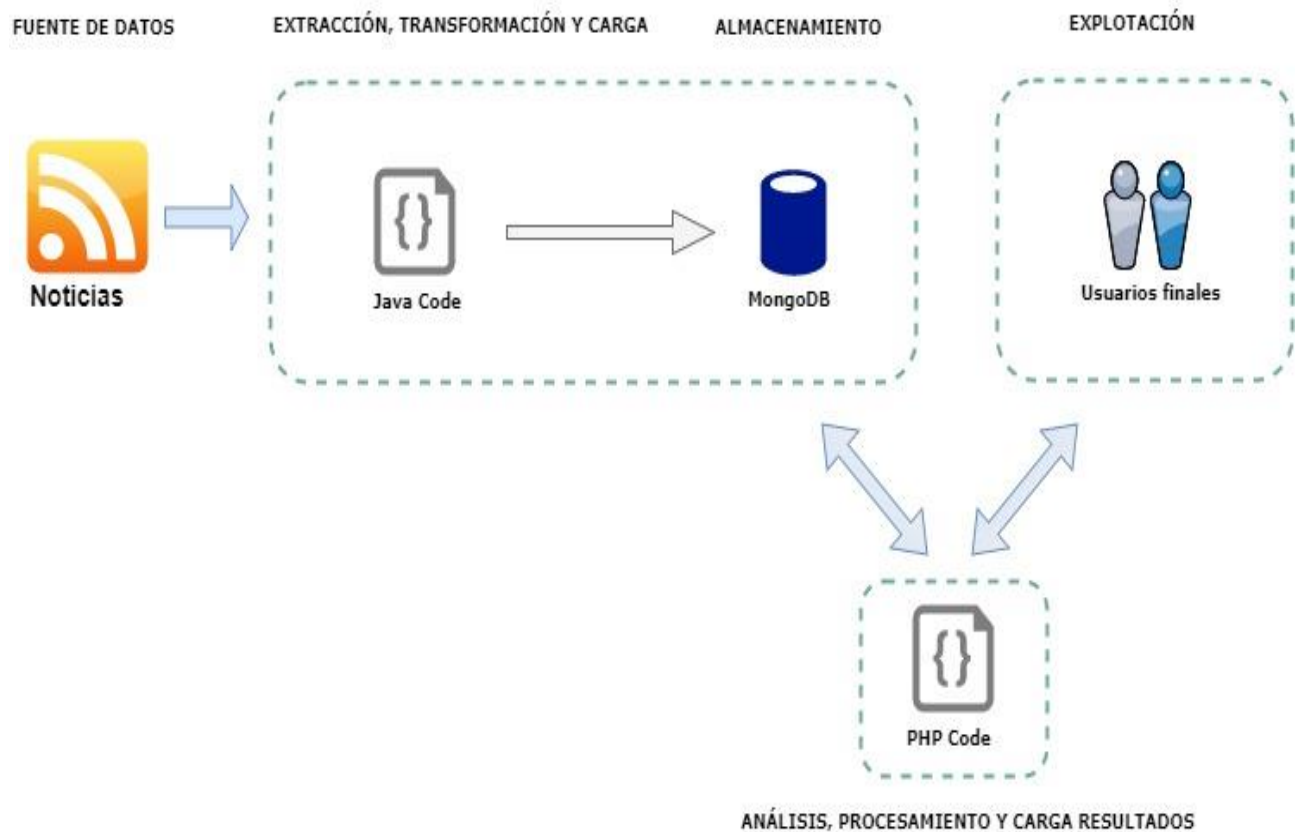


Figura 1.1. Capas tecnológicas.

Como vemos el código php es uno de los componentes más importantes pues se encarga de la gestión de la base de datos, análisis y procesamiento de los datos a través de distintos métodos y la carga de resultados para su posterior explotación.

1.1.2 Características

1.1.1.2.1 Volumen de información

El volumen se refiere a las cantidades masivas de datos que se almacenan con la finalidad de procesar dicha información, transformando los datos en información.

1.1.1.2.2 Velocidad de los datos

La velocidad se refiere a los datos en movimiento por las constantes interconexiones que realizamos, es decir, a la rapidez en la que son creados, almacenados y procesados en tiempo real.

1.1.1.2.3 Variedad de los datos

La variedad se refiere a las formas, tipos y fuentes en las que se registran los datos. Estos datos pueden ser datos estructurados y fáciles de gestionar como son las bases de datos, o datos no estructurados.

1.1.1.2.4 Veracidad de los datos

Se refiere a la incertidumbre de los datos, es decir, al grado de fiabilidad de la información recibida. Es necesario analizar los datos y determinar cuál es confiable y cuál no.

1.1.1.2.5 Valor de los datos

Los datos por sí solos no constituyen ningún valor, el valor se obtiene de datos que se transforman en información a través de los métodos oportunos.



Figura 1.2. Conceptos relacionados con BigData.

Capítulo 2.

Historia del BigData

En este capítulo se presenta de forma muy resumida los antecedentes y la actualidad en el ámbito del Big Data.

2.1 Antecedentes

En sus comienzos el tratamiento de datos que se llevaba a cabo se realizaba con cantidades de datos proporcionalmente pequeños por lo que con los sistemas convencionales era posible realizar el análisis de los mismos.

Sin embargo, el crecimiento del sector tecnológico, aparición entre otros muchos aspectos de las redes sociales, comercio electrónico, etc. provocó el aumento desmesurado del volumen de datos los cuales no podían ser tratados con los sistemas que se disponían antiguamente dando lugar al Big Data.

2.2 Actualidad

Actualmente las tecnologías de Big Data presentan un uso cada vez más elevado. Esto es debido a distintos factores. Por un lado, la reducción en la en los costes de esta tecnología, gracias al amplio conjunto de herramientas de código abierto que implementan los avances en este campo, y a la disponibilidad cada vez más asequible a plataformas de Cloud Computing que facilitan el acceso a grandes conjuntos de máquinas en modo de servicio o alquiler dinámico, pagando sólo por el gasto en recursos.

A esto se une también el auge de conceptos como Internet of Things, que permite la proliferación de todo tipo de dispositivos capturando y compartiendo datos para analizar diversos procesos en diferentes campos de aplicación. Todo lo anterior se une a la enorme repercusión mediática que adquieren muchas de las innovaciones tecnológicas generadas.

Además de esto, el Big Data está constituyéndose como un elemento fundamental con el que obtener ventajas competitivas para las empresas, obteniendo con ello grandes ganancias.

Capítulo 3.

Planificación y objetivos del proyecto

En este apartado se describen los pasos seguidos en la ejecución del proyecto, así como los objetivos que se persiguen con cada uno de ellos. La planificación del proyecto se basa en los 4 apartados siguientes.

3.1 Gestión de RSS

La gestión de los RSS hace referencia al proceso de obtención de estos ficheros desde la versión digital de un periódico, así como al tratamiento posterior de los mismos. Por tanto, la idea es la de obtener estos ficheros los cuales presentan un formato XML y eliminar todos los caracteres sobrantes (etiquetado, caracteres extraños, etc.) del mismo para así quedarnos únicamente con la información útil de cada noticia.

3.2 Creación de la base de datos

Al igual que el apartado anterior y cada uno de los que aquí se describen, este paso es fundamental en el desarrollo del proyecto. Se basa en la creación de la base de datos MongoDB que nos permitirá almacenar todos los datos de las noticias que hemos obtenido en el paso previo. Además, también se incluyen en este apartado la introducción de los datos de las noticias, así como de los lugares de las zonas de Santa Cruz de Tenerife y San Cristóbal de La Laguna.

Por tanto, se crearán dos colecciones, una para almacenar las noticias y otra para almacenar las posibles ubicaciones de esas noticias. Cada una de estas colecciones tendrá una estructura determinada:

- Noticias: La estructura de la colección de noticias estará determinada por los campos que tengan las noticias, siendo el número de campos los atributos de la colección. Además de los propios campos de las noticias también se añadirán tres atributos más que indiquen la ubicación y las coordenadas latitud y longitud de la ubicación obtenida para esa noticia.
- Lugares: La estructura de esta colección es mucho más simple que la anterior y solo constará de tres atributos que serán el *lugar* (ya se de Santa Cruz o La Laguna) y sus coordenadas (*latitud* y *longitud*).

También incluimos aquí la creación de todos los métodos necesarios para la gestión de la base de datos, además de otros métodos para la manipulación de las noticias.

3.3 Generación del mapa

Esta fase se centra en la creación del mapa. Para conseguir esto previamente debe haberse creado un fichero JSON (fichero con el que se va a trabajar para crear el mapa) que contenga toda la información relativa a las noticias.

La generación del mapa se hará haciendo uso del api de Google y este fichero JSON.

Esta fase tiene por objetivo, por tanto, la implementación de una parte de la estructura de la aplicación web y sobre todo la creación e inclusión del mapa (con algunas noticias) en ella.

3.4 Diseño de la aplicación web

Con este último apartado completado se dará por terminada la creación de la aplicación web. Para ello previamente se realizarán las siguientes acciones:

- Rellenar el mapa con todas las noticias que estén almacenadas en la base de datos.
- Personalizar el mapa a través de los propios métodos que proporciona el Api de Google.
- Terminar la estructura del servicio, así como el diseño del mismo.
- Añadir animaciones y eventos que hagan más simple la navegación por el sitio web.
- Añadir funcionalidad extra a la aplicación web que pueda ser de utilidad.
- Comprobar el correcto funcionamiento del servicio una vez esté terminado.

Por último, cabe mencionar que dividiendo el proyecto en estos apartados o fases se pretende conseguir, además de los objetivos específicos de cada uno de los apartados anteriores, un objetivo general que consiste en estructurar y organizar el proyecto de tal forma que su comprensión y sobre todo su implementación y puesta en marcha sean mucho más sencillas.

Capítulo 4.

Herramientas utilizadas

En este capítulo comentaremos las diferentes herramientas y tecnologías utilizadas durante el desarrollo del trabajo.

4.1 Tratamiento de RSS

Para el desarrollo del proyecto se ha decidido hacer uso de RSS que son ficheros que contienen una versión específica de las noticias publicadas en la web del periódico en cuestión. Además, el RSS se reescribe automáticamente cuando se produce alguna actualización en los contenidos del sitio web facilitando con ello la tarea de mostrar en el mapa de la aplicación web las noticias de forma actualizada.

Para la gestión y tratamiento de estos ficheros RSS se ha hecho uso del lenguaje Java y diferentes librerías.

4.1.1 RSS

Un fichero RSS presenta un formato XML que se utiliza para difundir información actualizada, frecuentemente a usuarios que se han suscrito a la fuente de contenidos. El formato permite distribuir contenidos sin necesidad de un navegador, utilizando programas llamados agregadores o lectores RSS de noticias, diseñados para leer contenidos RSS.



Figura 4.1. RSS

I.4.1.1.1 SAX

La API Simple para XML (SAX) es un estándar público desarrollado para el análisis basado en eventos de documentos XML. Este parseador es bastante útil puesto que requiere muy poca memoria para procesar los documentos XML, además, lee el documento de forma secuencial, obteniendo en cada momento sólo la parte que está siendo leída. Para poder manipular los datos que se van leyendo es necesario definir los ‘handlers’ que deben ser definidos por el programador.

4.1.2 Java

Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Su elección es básicamente porque es un lenguaje multiplataforma y muy útil en aplicaciones cliente/servidor como es el caso de este proyecto.

4.2 Base de datos

En lo que a la base de datos se refiere, como vemos a continuación, se ha decidido usar una base de datos no relacional puesto que son base de datos que permiten mayor escalabilidad y rendimiento en tiempo real que las tradicionales bases de datos relacionales. Además, la flexibilidad y rapidez que proporciona para los datos este tipo de BBDD es ideal para el desarrollo del proyecto.

4.2.1 MongoDB

MongoDB es NoSQL y guarda estructuras de datos en documentos similares a JSON con un esquema dinámico (MongoDB utiliza una especificación llamada BSON), haciendo que la integración de los datos en ciertas aplicaciones sea más fácil y rápida.

4.2.2 PHP

El lenguaje PHP se ha elegido para llevar a cabo la programación del código encargado de gestionar las peticiones a la base de datos, así como para implementar las respuestas del servidor a las peticiones del cliente web.



Figura 4.2. MongoDB y PHP.

4.3 Alojamiento Web

El alojamiento de la aplicación web se ha llevado a cabo en el servicio de la Universidad de La Laguna, IAAS, utilizando Apache como servidor

4.3.1 Apache

Apache es un servidor web de código abierto, el cual se puede utilizar perfectamente en sistemas operativos Linux, además de otros como Windows o Macintosh. Es un proyecto que cuenta con una gran comunidad e información, y una gran capacidad y eficacia de alojar diferentes servicios web.

Está optimizado para el uso del lenguaje PHP que será usado en el proyecto como ya se mencionó previamente. Además, es muy configurable, pero hace falta tener algunos conocimientos de administración de sistemas Linux.

4.3.2 IAAS ULL

Un IAAS (Infrastructure as a Service), es un servicio de infraestructura que permite acceso en la nube a diferentes recursos informáticos. Este tipo de infraestructuras permite al programador mayor control que otras infraestructuras similares para gestionar los recursos que haya en ella.

Es por esto por lo que para el despliegue de la aplicación se haya hecho uso de la maquina IAAS ULL que pone a disposición del alumnado la Universidad de La Laguna. La máquina en concreto solicitada cuenta con el sistema operativo Ubuntu 16.04, elegido expresamente para la utilización de la versión 7 de PHP, así como los drivers de MongoDB correspondientes.



Figura 4.3. Apache.

4.4 Diseño Web

En lo referente al diseño web se han empleado herramientas bastante habituales en este ámbito, así como otras esenciales para llevar a cabo tareas específicas.

4.4.1 Bootstrap

Este tipo de framework CSS es uno de los más usados actualmente y facilita bastante las tareas de redimensionado de una aplicación web, así como el diseño general.

4.4.2 HTML5, CSS3, JavaScript, JQuery

Todas ellas son básicas en el desarrollo de un entorno web. Por tanto, se ha utilizado HTML5, para crear la estructura y contenido del servicio. Además, es un lenguaje coordinado por la W3C (World, Wide, Web Consortium), regulador de la navegación web.

Para definir y crear la presentación del documento estructurado escrito en HTML5, se ha utilizado el lenguaje CSS3, que corresponde a la última versión de este lenguaje.

Por último, para la interacción del usuario con el entorno web y para la gestión de los eventos de la interfaz, las animaciones y el árbol DOM se han empleado los lenguajes JavaScript y JQuery respectivamente

4.4.3 JSON

Es un formato de texto ligero para el intercambio de datos que puede ser leído desde cualquier lenguaje de programación. Este tipo de fichero nos permitirá transferir datos usados por la aplicación web de forma simple y rápida.

Además, su sencilla estructura facilita la tarea del programador a la hora de gestionar dichos ficheros.



Figura 4.4. JSON.

4.4.4 FPDF

FPDF es una clase PHP que permite generar archivos PDF con PHP puro, es decir, sin usar la biblioteca PDFlib. Esta clase cuenta con números métodos que facilitan la creación del pdf y que permite la generación de un pdf totalmente personalizado.

4.5 Mapa

4.5.1 Google Maps JavaScript API

Para la generación del mapa se ha decidido usar el API de Google que ya cuenta con numerosos métodos que facilitan bastante la creación e integración de un mapa en una aplicación web como la que se ha desarrollado. Además de esto permite la personalización de los mapas generados, visualización de datos procedentes de ficheros con distintos formatos aceptados, generación de formas en el mapa, etc.

Capítulo 5.

Aplicación Web

En este capítulo se detallan y describen cada una de las implementaciones llevadas a cabo para la consecución del entorno web final, así como los resultados obtenidos una vez se ha completado la aplicación.

5.1 Arquitectura de la aplicación

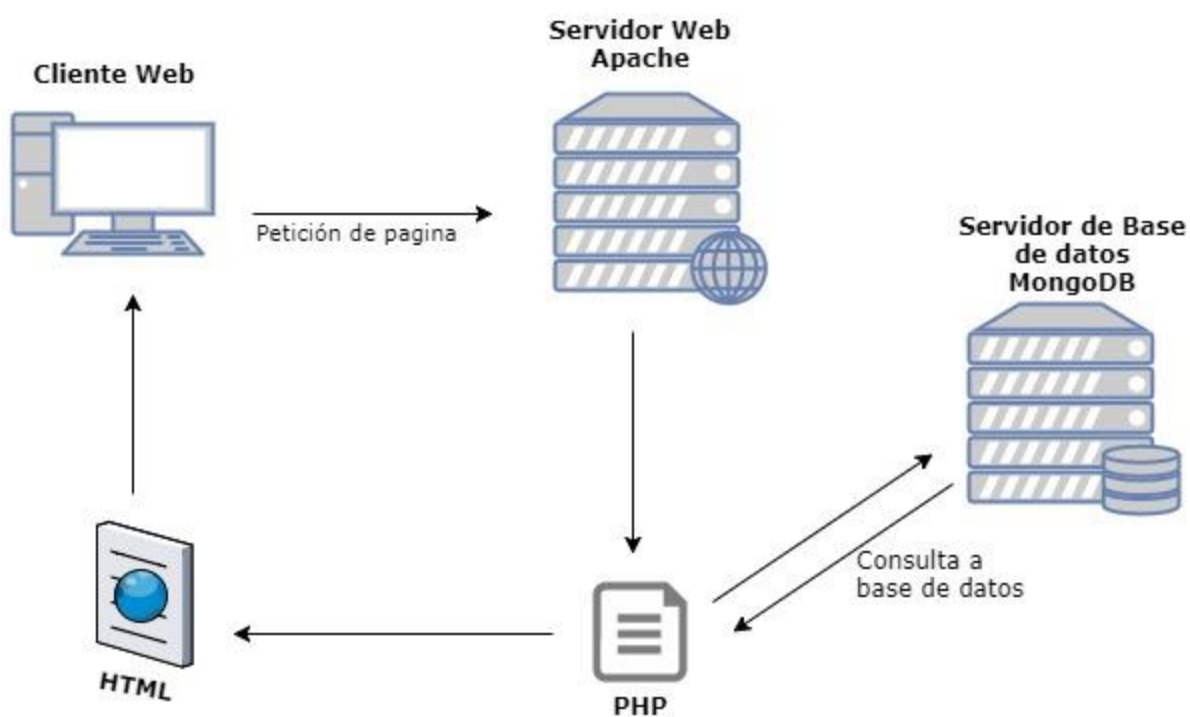


Figura 5.1. Arquitectura aplicación.

5.2 Estructura de directorios

Este apartado se muestra la estructura de directorios de la que consta la aplicación y se describe la finalidad de cada uno de ellos:

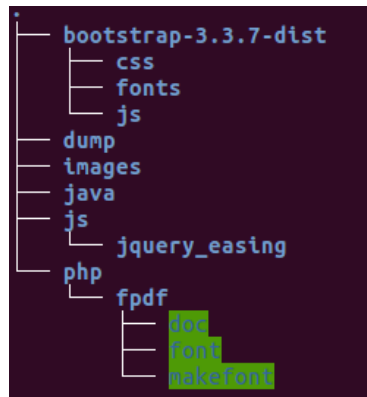


Figura 5.2. Estructura de directorios de la aplicación.

Como se ve la imagen la aplicación cuenta con varios directorios:

- **bootstrap-3.3.7-dist:** Directorio para el contenido del framework
- **dump:** Contiene el backup de las colecciones y ficheros de descarga
- **java:** Contiene todo el código desarrollado en java
- **php:** Ficheros php encargados de gestión de base de datos y otras labores.
 - **fpdf:** Destinado a almacenar el código fuente de la clase php FPDF usada para la generación de pdf.
- **js:** Contiene el código JavaScript propio de la aplicación para, entre otras, crear el mapa.
 - **jquery_easing:** Directorio del plugin jquery_easing.js.
- **images:** Contiene todas las imágenes usadas en la aplicación web.

5.3 Implementación

En este apartado vamos a detallar las funciones más importantes creadas para completar cada una de las fases mencionadas [previamente](#), así como la finalidad de cada una de ellas.

5.3.1 Gestión de RSS

Para la gestión de los RSS que contienen las noticias se decidió emplear el lenguaje Java como ya se mencionó en apartados anteriores. A continuación, se describirán las distintas clases empleadas para esta finalidad, así como también se mencionarán algunos de los métodos empleados para llevar a cabo esta tarea:

MongoClienteNoticia.java

Este es el cliente principal, es la clase encargada de crear la conexión a MongoDB y gestionar las consultas. Cuenta por tanto con un constructor encargado de establecer la conexión con mongo:

```
/**
 * Clase para crear una conexión a MongoDB.
 * @return MongoClient conexión
 */
public MongoClienteNoticia() {
    this.mongo = new MongoClient("localhost", 27017);
    db = mongo.getDB(DATABASE_PUBLIC);
    table = db.getCollection(TABLE_NOTICIAS);
}
```

Figura 5.3. Constructor cliente MongoDB.

Además de este, cuenta con otros métodos utilizados para realizar conexiones básicas y consultar o insertar datos en la base de datos. Estos métodos son:

- ***printDatabases()***: Imprime por pantalla todas las base de datos MongoDB.
- ***mostrarTabla()***: Muestra una colección y cuenta el número de documentos que contiene.
- ***Find()***: Busca una noticia en la colección ‘noticia’. Devuelve true si encuentra la noticia o false si no es así.

- ***findObject(string titular)***: Busca una noticia en la base de datos en la colección ‘noticia’. La búsqueda se realiza utilizando el método anterior y un query en el que se especifica el campo y el valor a buscar. Una vez encontrado se crea un objeto noticia con los valores de esa noticia. Devuelve un objeto noticia si la noticia es encontrada o null en caso contrario.
- ***insertNoticia(Noticia n)***: Inserta una noticia en la base de datos. Para realizarlo primero se comprueba si ya existe la noticia con el método find() mencionado anteriormente, y en caso de no encontrarla entonces se crea un nuevo documento con los campos y los valores correspondientes que contiene la noticia. Finalmente se inserta en la base de datos mediante el método insert que provee la clase DBCollection. Devuelve true una vez se inserta en la base de datos.
- ***borrarNoticia(String titular)***: Borra una noticia identificada por su titular.
- ***borrarALL()***: Borra una colección y todo su contenido.

RSS.java

Esta clase se encarga únicamente de crear el RSS para ello cuenta con un constructor básico que recibe como parámetros la *url* al sitio en el que se encuentra el rss, el *tipo* de rss y el *periódico* al que pertenece como vemos en la imagen siguiente:

```
Rss(String url, String tipo, String periodico) {
    this.url = url;
    this.tipoRss = tipo;
    this.periodico = periodico;
}
```

Figura 5.4. Constructor de rss.

Un ejemplo de uso del constructor es el que vemos a continuación donde se crea un vector de RSS de tal forma que podamos cargar varios ficheros.

```
public static Rss[] vectorRss = {
    new Rss ("http://eldia.es/rss/santacruz.rss", "Santa Cruz de Tenerife", "El DÃ-a"),
    new Rss ("http://eldia.es/rss/sociedad.rss", "Sociedad", "El DÃ-a Sociedad"),
    new Rss ("http://eldia.es/rss/sucesos.rss", "Sucesos", "El DÃ-a Sucesos"),
    new Rss ("http://www.laopinion.es/elementosInt/rss/98", "Santa Cruz de Tenerife", "La OpiniÃ³n")
};
```

Figura 5.5. Vector de rss.

Noticia.java

Esta clase representa a una noticia y al igual que las anteriores consta de un constructor, sin embargo, en este caso se trata de un constructor sobrecargado. De esta forma podremos crear una noticia de dos maneras distintas como vemos a continuación:

```
public Noticia (Rss rss, Element node) throws ParseException {
    this.titular = Utils.tratar(node.getChildText(Rss.TITLE));
    this.descripcion = Utils.tratar(node.getChildText(Rss.DESCRIPCION));
    this.enlace = node.getChildText(Rss.LINK);
    this.tipo_noticia = rss.tipoRss;
    this.periodico = rss.periodico;
    this.fecha = extraerFecha (this.enlace);
}

public Noticia (String titular, String descripcion, String enlace, String tipo_noticia, String periodico, Date fecha) {
    this.titular = titular;
    this.descripcion = descripcion;
    this.enlace = enlace;
    this.tipo_noticia = tipo_noticia;
    this.periodico = periodico;
    this.fecha = fecha;
}
```

Figura 5.6.Constructor sobrecargado de noticias.

- En el primer caso se construye el objeto partiendo de un RSS para lo cual se usa la clase ‘Element’ de java de tal forma que usando el método getChildText() se puedan obtener los campos del RSS correspondientes al título, descripción y link de la noticia.

CapturarRSS.java

Esta clase se encarga de leer el rss y guardar las noticias leídas en una lista llamada “listaNoticiasClasificadas”. Consta de dos métodos:

- **obtenerInserts():** Este método básicamente lo que hace es recorrer el vector de RSS y va pasando cada uno de los RSS a la función siguiente.
- **obtenerInserts(Rss rss):** Este método recibe el RSS pasado desde la función anterior y mediante la librería externa sax, genera un grafo del XML (formato del fichero RSS). El método se encarga de recorrer dicho grafo obteniendo el titular, descripción y link de la noticia para pasar los campos a al método que necesita el cliente de base de datos.

Utils.java

Esta clase como contiene métodos útiles para el resto de clases, es decir, métodos que son usados en el resto de clases. Uno de ellos es el siguiente:

- ***String tratar (String in)***: Este método se utiliza para reemplazar caracteres extraños por sus equivalentes y que de esta forma el string pasado por parámetro tenga el formato adecuado. Es usado sobre todo en los títulos y descripciones de las noticias.

5.3.2 Creación y gestión de la base de datos

En este apartado se comienza el desarrollo en PHP, implementando el fichero **mongodb.php** el cual es el encargado de llevar a cabo las conexiones a la base de datos y los métodos para gestionar la misma de lado del servidor.

A continuación, se procederá a explicar los métodos de los que consta:

- ***get_noticia(\$id)***: Esta función se encarga de obtener una noticia en concreto a partir del id de la misma. Devuelve un array con los distintos campos de la noticia y sus valores.
- ***get_all_noticias(\$coleccion, \$limit=0)***: Permite obtener la información de un número determinado de noticias de una colección en concreto. Devuelve un array de arrays con todas las noticias encontradas. Consta de dos parámetros:
 - *\$coleccion* que indica la colección de la cual obtener las noticias
 - *\$limit* que permite establecer el máximo de filas a devolver en la consulta.
- ***count_all_noticias()***: Permite contar todos los documentos de la colección de noticias. El método devuelve el número correspondiente a la cuenta realizada.
- ***remove_coleccion(\$coleccion)***: Permite eliminar una colección completa de la base de datos. La colección por eliminar será la indicada mediante el parámetro *\$coleccion*.

- ***remove_noticia(\$id, \$coleccion)***: Elimina una noticia identificada por su *\$id* de la colección indicada con *\$coleccion*.

Como vemos en estos métodos no se encuentra un método insert para insertar noticias en la base de datos puesto que de ello se encargaría el código desarrollado en java que hemos visto anteriormente.

Además de este fichero php se han creado otros dos ficheros que contienen métodos usados para insertar los lugares de las zonas de Santa Cruz y La Laguna en la base de datos, calcular la ubicación de cada noticia, así como otros métodos útiles en el desarrollo.

utils.php

De la misma forma que en el código en Java la finalidad del fichero utils es la de proveer métodos generales que permitan completar la implementación de otros. En el encontramos las funciones siguientes:

- ***quitar_tildes(\$cadena)***: Como su nombre indica su finalidad es la de eliminar acentos y sustituir caracteres extraños por otros básicos. Esta función es usada para limpiar los textos y reducir por ende la complejidad en el análisis de los mismo.
- ***delete_caracteres_sobrantes(\$texto)***: Este método es usado para el cálculo de la ubicación de la noticia. Su función es la de eliminar caracteres que no son necesarios analizar para buscar la ubicación de la noticia, quedándonos únicamente con las palabras que tienen más probabilidad de ser la ubicación. Para ello se hace uso de diferentes expresiones regulares.

El método seguido con las expresiones regulares es el de eliminar cualquier símbolo extraño que puedan contener los textos, eliminar acentos y por último quedarnos únicamente con las palabras escritas en mayúsculas (simples y compuestas).

Esto es así, puesto que se entiende que los nombres de los lugares (ubicaciones) son nombres propios que deben ir en mayúscula. Con ello se reduce bastante el campo de búsqueda y por tanto se disminuye la complejidad del proceso de cálculo.

- ***split_array(\$array)***: Este es un método bastante útil y muy importante dentro del desarrollo global de la aplicación. La finalidad del método es la de dividir el array de noticias (*\$array*), agrupando las noticias en base a su ubicación. De esta forma tendremos un array final en donde en cada posición tendremos todas las noticias pertenecientes a una misma ubicación.

La decisión de implementar este método y proponer esta solución es, como veremos más adelante, para facilitar la generación del mapa.

funciones.php

En este fichero nos encontramos con los métodos más importantes del código desarrollado en php. El más importante de todos ellos sin duda alguna es el encargado de obtener la ubicación de cada noticia. A continuación, se describen cada uno de ellos:

- ***insert_ubicacion()***: Esta función se encarga de calcular la ubicación, si la hubiera, de la noticia. Para ello se hace una consulta a la base de datos recuperando de ella las noticias y los lugares almacenados, guardando en un array los lugares y sus ubicaciones.

A continuación, se realizan distintas comparaciones entre las palabras claves de la noticia y los lugares que hemos almacenado en el array tal y como se ve en la imagen siguiente:

```

//BUSCAR EN EL TITULO alguna coincidencia con los lugares
if($encontrado == false){
    foreach($titulo as $t){
        foreach($barr as $p){
            if (in_array($t, $p)) { //Comparacion de palabra exacta (en mayuscula)
                $ubicacion = $p[0];
                $latlong = [$p[1],$p[2]]; //latlong[0] -- latitud | latlong[1] -- longitud
                $encontrado = true;
                break;
            }
        }
        if($encontrado==true)
            break;
    }
    if($encontrado == false){
        foreach($barr as $p){
            if(preg_grep("/\b".$p[0]."\b/", $titulo) ){ //Comparacion parcial (en mayuscula)
                $ubicacion = $p[0];
                $latlong = [$p[1],$p[2]];
                $encontrado = true;
                break;
            }
        }
    }
}
}

```

Figura 5.7. Comparaciones para el cálculo de la ubicación.

Como vemos en la imagen anterior, en este caso, se está buscando la ubicación de la noticia en el título de la misma. Para ello se realizan dos tipos de comparaciones:

- Una **comparación exacta** que consiste en comparar cada una de las palabras claves obtenidas del texto del título con los nombres de los lugares almacenados en la base de datos.
- Una **comparación parcial** que consiste en comparar cada una de las palabras claves obtenidas del texto del título con una parte de los nombres de los lugares almacenados. Es decir, si una de las palabras claves encontrada en el texto fuera “Igueste” y en la base de datos tenemos almacenado el nombre con la forma “Igueste de San Andrés”, esta comparación haría que ese nombre sea detectado como la ubicación a diferencia de la comparación anterior.

En caso de que no se haya encontrado la ubicación en el título se realiza exactamente el mismo procedimiento, pero esta vez recorriendo la descripción de la misma.

Finalmente se actualiza la base de datos añadiendo a cada una de las noticias la ubicación obtenida:

```
//GUARDAR UBICACION EN LA BBDD CON SUS COORDENADAS
if($encontrado == true){
    $GLOBALS['noticias_ubicadas']++;
    $bulk->update(
        ['_id' => new MongoDB\BSON\ObjectId($n->_id)],
        ['$set' => ['ubicacion' => $ubicacion, 'latitud' => $latlong[0], 'longitud' => $latlong[1] ]],
        ['multi' => false, 'upsert' => false]
    );
    //echo "La ubicacion es: ".$ubicacion." Latitud: ".$latlong[0]." Longitud: ".$latlong[1]."<br>";
}
elseif($encontrado == false){
    $bulk->update(
        ['_id' => new MongoDB\BSON\ObjectId($n->_id)],
        ['$set' => ['ubicacion' => "No encontrada", 'latitud' => 0, 'longitud' => 0 ]],
        ['multi' => false, 'upsert' => false]
    );
}
}
$mongo->executeBulkWrite('NoticiasDB.noticia', $bulk); //Actualizar el campo ubicacion de la coleccion de noticias de la base de datos
```

Figura 5.8. Actualización de la ubicación de las noticias.

En referencia a este procedimiento se ha realizado una prueba del software con 30 noticias para determinar cuáles son ubicadas. Esta prueba se encuentra en el [Apéndice 1](#) de este documento.

- ***insert_coordenadas_bd(\$archivo)***: Este método se ha implementado para rellenar la colección de coordenadas (colección que contiene los lugares de las zonas analizadas) a partir de ficheros csv. Por tanto, el método abre cada uno de los ficheros csv que estén dentro del directorio dump y va guardando en un array de arrays el contenido que haya en él.

Posteriormente se recorre dicho array de registros y se van introduciendo cada uno de ellos en la colección de coordenadas de la base de datos como vemos a continuación:

```

for ($i = 0; $i < count($registros); $i++) {

    $lugar = preg_replace("/[^A-Z0-9Ñ\-.]/", " ", strtoupper(quitar_tildes($registros[$i]['NOMBRE'])));
    $lugar = preg_replace("/\s+/", " ", $lugar);
    $lugar = trim($lugar);

    $doc = array(
        'id'           => new MongoDB\BSON\ObjectID,      #Generate MongoID
        'LUGAR'       => $lugar,
        //'DISTRITO'   => $registros[$i]['DISTRITO'],
        'LATITUD'     => $registros[$i]['GRAD_Y'],
        'LONGITUD'    => $registros[$i]['GRAD_X']
    );
    $bulk->insert($doc);
}

```

Figura 5.9. Inserción de coordenadas por medio de csv

- ***create_json_marcadores()***: Este método se encarga de la creación de un fichero json que será usado para la generación del mapa. Este fichero json se construye a partir de las noticias almacenadas en la base de datos, donde cada una de ellas ya tiene la ubicación calculada.

Además, y como mencionábamos anteriormente, se hace uso del método ***split_array()*** para agrupar las noticias con ubicación igual en una misma posición del array final.

Finalmente se crea el fichero si no existe o se sobrescribe si ya existe, codificando el array como json a través del método ***json_encode()*** de php.

- ***limpiar_csv(\$archivo)***: Este es un método que no se usa actualmente puesto que está pensado para los casos en los que se desea añadir lugares o zonas de Santa Cruz de Tenerife o La Laguna a partir de un fichero csv. La finalidad del método es eliminar las columnas irrelevantes de dicho csv quedándonos únicamente con las columnas referentes al nombre del lugar y sus coordenadas x e y.

5.3.3 Generación del mapa

Para abordar este problema se ha empleado Google Maps Api JavaScript junto con código desarrollado en JavaScript y jQuery.

Todo este código se encuentra dentro del fichero **index.js** junto con otros métodos que son usados en la aplicación web.

Inicialización del mapa

Dentro de este fichero está el método *init_map()* que como su nombre indica se encarga de inicializar el mapa y crear los objetos necesarios para su creación:

```
function initMap() {  
  
    map = new google.maps.Map(document.getElementById('map'), {  
        center: {lat: 28.48, lng: -16.32}, //(Y,X)la laguna  
        zoom: 11,  
        mapTypeControl: false,  
        fullscreenControl: false  
    });  
  
    infoWin = new google.maps.InfoWindow();
```

Figura 5.10. Inicialización de mapa

Posteriormente, mediante JQuery se obtiene el fichero json de noticias que creamos con el código php mencionado en la fase anterior (método *create_json_marcadores*) y se recorre dicho json creando contenido personalizado para los infowindow de cada marker del mapa, así como añadiendo a un array todas las noticias.

Añadiendo marcadores

Una vez hecho se llama al método *add_markers()* para que se añadan los marcadores oportunos al mapa.

El método recibe el array *locations* (contiene las noticias), el objeto *map* y una variable global denominada *infoWin* que es un objeto de la clase *InfoWindow* de Google Maps.

Este método usa el método *map* de arrays para recorrer el array 'locations' y crear un marcador por cada grupo de noticias que haya en el array.

Como habíamos mencionando en un apartado previo las noticias están agrupadas según ubicación, esto es así, para crear únicamente un marcador por cada posición distinta del array ‘locations’, es decir, un marcador por cada grupo de noticias pertenecientes a la misma ubicación.

Por último, cada marcador creado se añade al clúster de marcadores. Este último paso se realiza para implementar la agrupación de marcadores en clústeres de tal forma que aquellos marcadores que estén próximos unos a otros se agrupen si el mapa está lo suficientemente alejado. De esta forma, cuando hagamos zoom en el mapa esos clústeres desaparecerán, apareciendo con ello los marcadores correspondientes a dicha agrupación de marcadores.

Búsqueda en mapa

Además de lo anterior se ha implementado un método de búsqueda para no tener que buscar manualmente haciendo zoom un lugar dentro del mapa.

Esta función hace uso del método `keyup` de jQuery de tal forma que por cada letra que escribamos en el campo de búsqueda se muestran los posibles lugares que contienen dicha letra en su nombre. Para realizar dicha comparación se hace un ‘match’ con cada el nombre de cada uno de los lugares almacenados en el json de noticias.

Todos los nombres que coincidan son añadidos a un array el cual es recorrido posteriormente para mediante el método `innerHTML` y `appendTo()` añadir el contenido html correspondiente que contiene los nombres de los lugares.

5.3.4 Diseño de la aplicación web

Para el diseño de la aplicación se ha utilizado Bootstrap dada la simplicidad que adquiere un entorno web cuando se emplea este framework. En este sentido existe un fichero principal denominado `index.php` en el cual está contenida la estructura del entorno web.

Para mejorar el aspecto visual de la aplicación y añadir nuevas funcionalidades que en fases previas no habían sido implementadas, se han creado los métodos que veremos a continuación.

cerrarLeyenda() y abrirLeyenda()

Estos dos métodos han sido creados con la finalidad de mostrar el mapa de forma más expandida que la forma inicial que tiene el mismo. A pesar de tener una finalidad distinta puesto que uno amplía el mapa y otro lo reduce, la implementación de ambos es bastante similar.

Las diferencias más importantes las observamos al final de ambos métodos:

```
//Redimensionar y centrar mapa
map.setZoom(12);
google.maps.event.trigger(map, "resize");
map.setCenter({lat: 28.48, lng: -16.32});
```

Figura 5.11. Ampliar mapa

```
//Redimensionar y centrar mapa
map.setZoom(11);
//var center = map.getCenter();
google.maps.event.trigger(map, "resize");
map.setCenter({lat: 28.48, lng: -16.32});
```

Figura 5.12. Reducir mapa

descargarNews() y descargarNewsTxt()

Estos métodos han sido creados para permitir al usuario que está usando la aplicación descargar las noticias de una ubicación en concreto en dos formatos distintos, json y txt.

Ambos métodos se realizan bajo el mismo procedimiento, el cual consiste en realizar un ajaxpost al fichero downloadfile.php pasando como parámetros el json de noticias, así como otras opciones utilizadas en el código php.

Dentro del fichero php mencionado se encuentra una primera parte donde se crea el fichero (json o txt) y se añade el contenido de esa ubicación en él y una segunda parte encargada de la descarga del fichero desde el servidor.

descargarNewsPdf()

Esta es otra de las funcionalidades añadidas, más centrada en un usuario normal, al que le pueda parecer interesante descargar las noticias de una ubicación en pdf de tal forma que le sea más sencillo consultar cada una de las noticias en lugar de hacerlo a través de la propia aplicación.

Para elaborar este método se ha hecho uso de una clase php denominada FPDF descrita [aquí](#). A esta clase se le han añadido dos métodos denominados header y footer para generar un header y un footer específico para todos los pdf.

De esta forma, cuando desde el fichero index.js se hace un ajaxpost a este fichero, se comienza a crear el pdf y se guarda una vez se haya completado.

Cuando el fichero es guardado se retorna al método que contiene el Ajax post el cual es el encargado de realizar la descarga/visualización del pdf desde el navegador.

5.4 Resultados

5.4.1 Resoluciones

A pesar de estar implementada usando el framework Bootstrap, la aplicación no es 100% responsiva debido a la inclusión del mapa y diferentes animaciones de las que consta.

De esta forma, la aplicación puede ser visualizada sin ningún tipo de problemas en pantallas superiores a 830 de ancho. El alto de la página no supone ningún problema, aunque es recomendable que sea superior a 500 para poder ver el contenido adecuadamente.

Por tanto, todas las resoluciones de pantalla con un ancho superior al indicado son totalmente óptimas para la visualización de la aplicación.

Ejemplos de resoluciones óptimas:

Resolución: 1920x1080

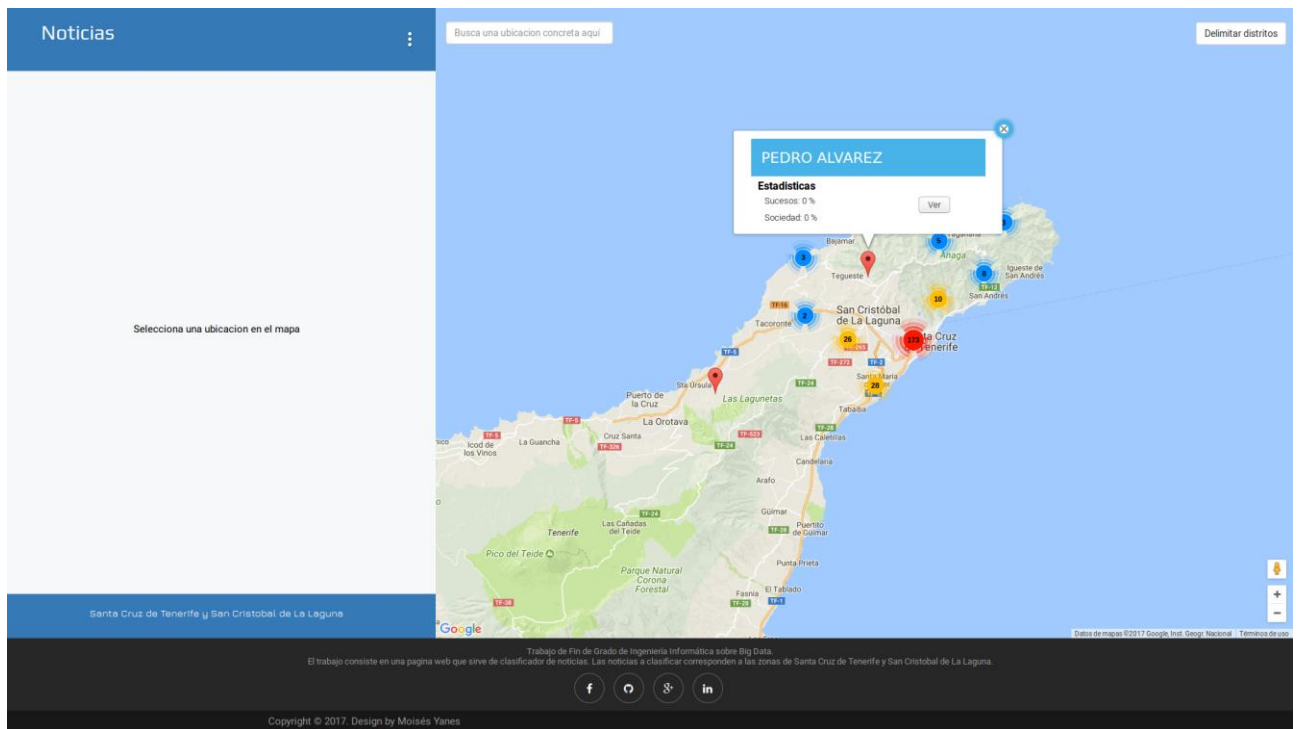


Figura 5.13. Resolución 1920x1080

Resolución: 1366x600

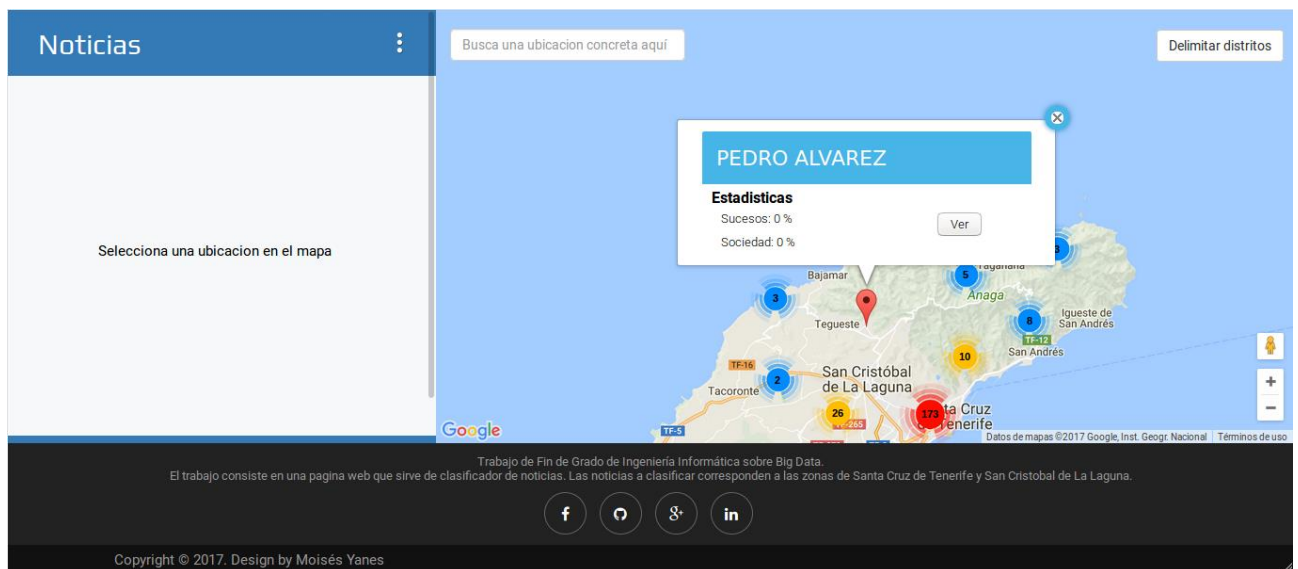


Figura 5.14. Resolución 1366x600

5.4.2 Presentación de resultados

En este apartado se muestran los resultados obtenidos, es decir, la aplicación web desarrollada, así como algunos de sus usos o funcionalidades.

El aspecto inicial de la aplicación una vez accedemos a ella es el que vemos en la imagen siguiente:

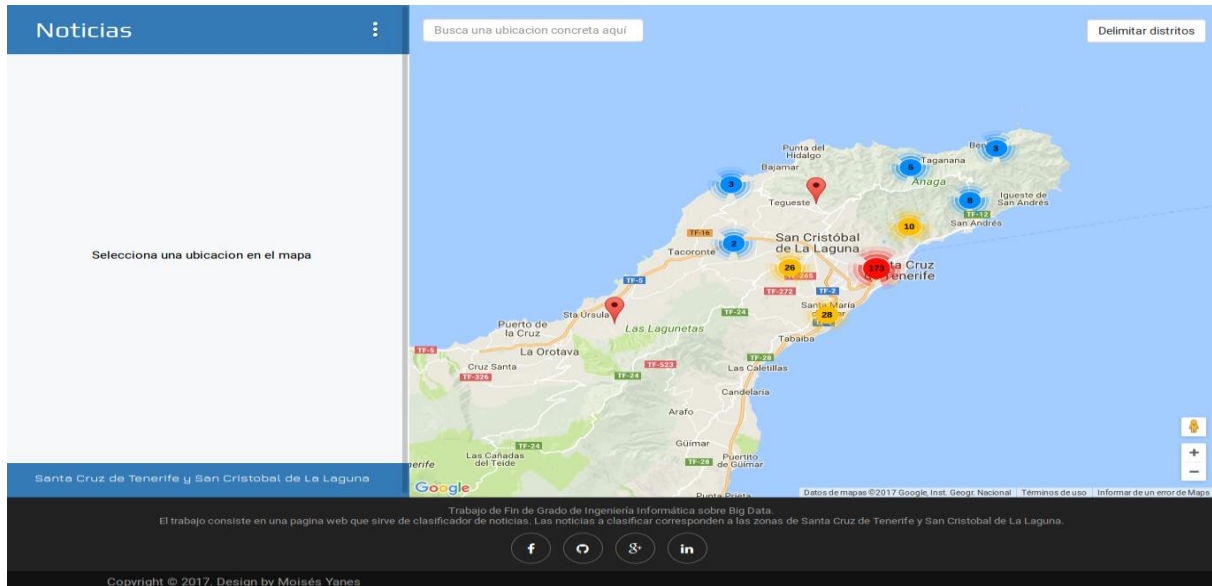


Figura 5.15. Vista inicial

Una vez dentro, si seleccionamos una ubicación en concreto se mostrará un panel informativo con algunas estadísticas y la opción de consultar las noticias de esa ubicación seleccionada:

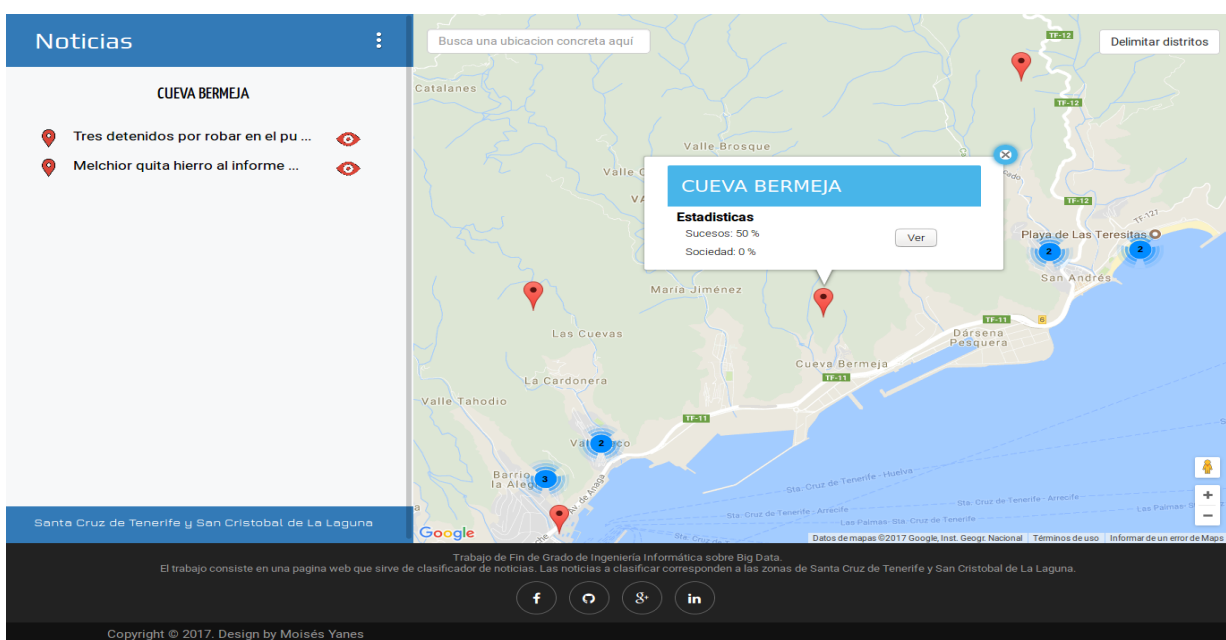


Figura 5.16. Vista de marcadores y noticias

Dentro de cada noticia podremos acceder a la información de la misma haciendo clic sobre el icono con forma de ojo que acompaña al título:

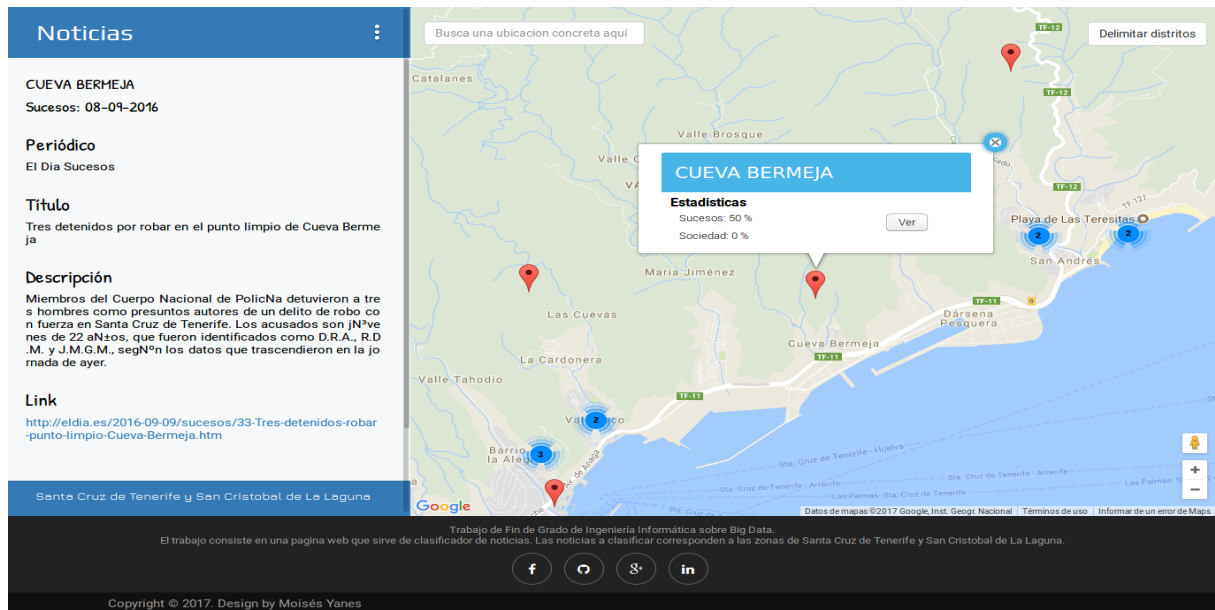
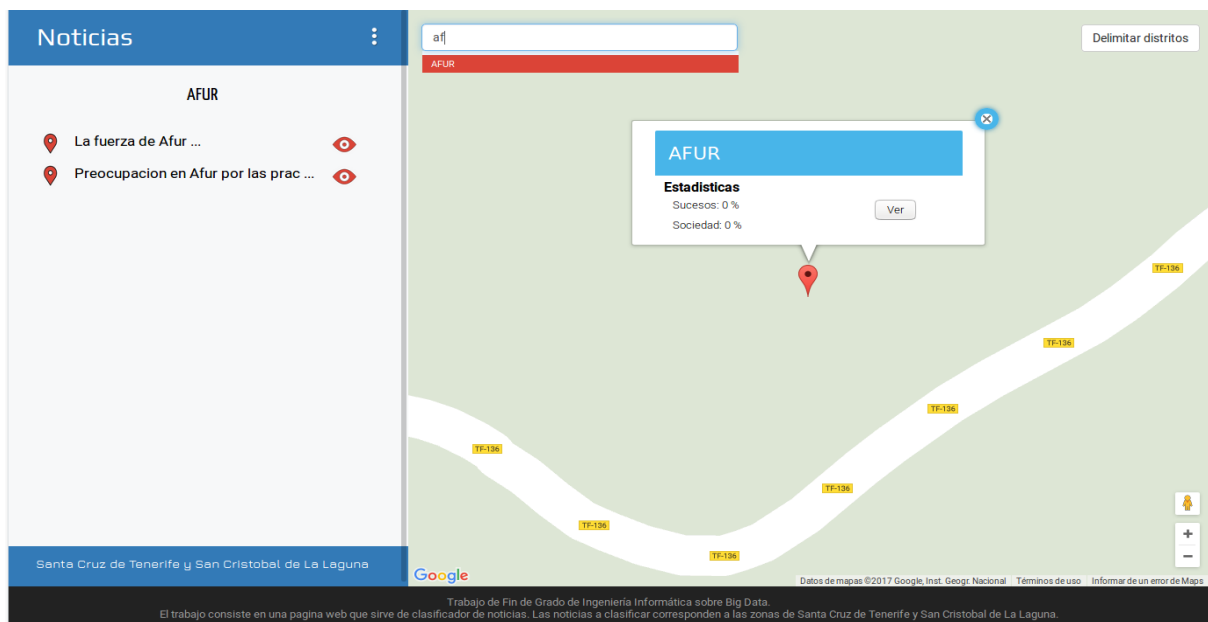


Figura 5.17. Visualización de información de noticias

Como funcionalidades extra, podemos buscar directamente la ubicación que queremos consultar en lugar de tener que navegar por el mapa. A medida que escribimos letras en el buscador este nos arroja ubicaciones que coinciden con esos caracteres introducidos, una vez salga el que buscamos bastará con clicar sobre él para que la página nos redirija a esa ubicación:



Otra funcionalidad añadida es la posibilidad de generar un pdf con las noticias de una ubicación concreta:

Listado de Noticias	
PERIODICO	El Dia
TITULAR	La fuerza de Atur
DESCRIPCION	Decir Gregoria Alonso Jimenez no supone nada especial. Ni siquiera en Atur. Pero nombrar a Goya "CaÁzona", como se la conoce cariÁzosamente, ya es otra cosa. No solo allí, en su pueblo, donde hace decadas que preside la asociacion de vecinos La Cumbrecilla, sino en todo Anaga. Por su dedicacion al Mazizo y por cuidar la ermita del caserío recibio este pasado fin de semana el merecido reconocimiento de sus paisanos en el marco de las fiestas patronales en honor de San Pedro.
LINK	http://eldia.es/2016-07-14/santaacruz/5-fuerza-Atur.htm
<small>UBICACION AFUR</small>	<small>Fecha: 13-07-2016</small>

Figura 5.18. Generación de PDF

Capítulo 6.

Problemas encontrados

En esta capítulo se comentarán algunos de los problemas encontrados durante el desarrollo de la aplicación. Se reflejarán los problemas que mayor dificultad han supuesto para el avance del proyecto y que por ende han conllevado mayor tiempo para solucionarse.

6.1.1 Eliminación de noticias de la base de datos

Este fue uno de los primeros grandes problemas con el que nos encontramos al comenzar el desarrollo. Una vez se habían insertado en la base de datos todas las noticias procedentes de los RSS se detectó que había noticias que no hacían referencia a las zonas de análisis de este proyecto, sino que eran noticias de carácter nacional, internacional u otros tipos.

Este tipo de noticias en ningún caso serán ubicadas puesto que no contienen en ellas referencias a las zonas de análisis, es por ello por lo que se decidió eliminar dichas noticias para reducir también con ello el proceso de análisis y cálculo de ubicación. Es en este punto donde surge el problema, y es que para eliminar todas esas noticias se precisó el uso de diversas expresiones regulares que permitieran eliminar la mayoría de estas noticias de una sola vez.

Sin embargo, esto no fue así puesto que había noticias que no casaban con dichas expresiones lo que conllevó una mayor inversión de tiempo en analizar cuáles de las noticias no estaban siendo eliminadas y por qué. Cabe destacar que este proceso con cantidades pequeñas de información es bastante simple, pero al tratarse de miles de noticias adquiere bastante complejidad.

Finalmente, tras un largo proceso de análisis y el empleo de las expresiones regulares directamente desde el MongoDB Shell se consiguió eliminar este tipo de noticias.

6.1.2 Generación de mapa

La creación del mapa fue otro problema importante dentro del desarrollo general de la aplicación. A pesar de que la Google Maps API JavaScript facilita bastante la generación de mapas en un entorno web como el nuestro, surgieron algunos problemas relacionados con la obtención de los datos necesarios para plasmar la información de las noticias sobre el mapa.

Tras las primeras investigaciones en la página de la API, así como en otras páginas se observó que la solución era bastante compleja puesto que para realizarla había que generar un documento XML a partir de una tabla de la base de datos donde tuviéramos únicamente los marcadores a pintar sobre el mapa y posteriormente cargar el XML en el mapa usando métodos como XMLHttpRequest. Además, la mayoría de ejemplos estaban destinados a gestores de base de datos relacionales como MySQL.

Descartada esta opción debido a la complejidad y el largo proceso que supondría su implementación se decidió buscar una solución más sencilla. La API de Google, además de proveer el método comentado anteriormente, también permite añadir contenido al mapa partiendo de un fichero JSON.

Es por ello por lo que se empleó esta solución creando un fichero JSON mediante el código php y consultas a MongoDB, guardando dicho fichero en el servidor. Posteriormente este fichero sería cargado desde el mapa importando en él la información que deseábamos.

6.1.3 Superposición de marcadores

Este otro gran inconveniente surgió después de haber creado el mapa y añadido el fichero json con los datos a cargar. El problema era que, al cargar los marcadores en él, aquellos que hacían referencia a la misma ubicación se superponían unos detrás de otros siendo imposible acceder a los que permanecían detrás.

Para resolver este problema se emplearon dos métodos distintos con la finalidad de conseguir que la visualización del mapa y de las noticias fuera lo más intuitiva y sencilla posible.

I.6.1.3.1 Agrupación de noticias por ubicación

Esta primera solución está basada en la agrupación de las noticias por ubicación, con esta solución se pretendía que se crearan únicamente tantos marcadores como ubicaciones distintas hubiese. De esta forma, ya no tendríamos marcadores sobrepuestos en una misma ubicación puesto que solo se crearía uno para cada ubicación.

I.6.1.3.2 Agrupación de marcadores en clústeres

La otra solución implementada para resolver el problema de los marcadores consistía en agrupar los marcadores más cercanos unos a otros en clústeres. Los clústeres de marcadores son agrupaciones de marcadores que, en nuestro caso, se agruparían por proximidad unos con otros.

Después de consultar la API se observó que ésta provee de una biblioteca denominada MarkerClusterer que permitía precisamente realizar esta tarea casi sin ninguna dificultad.

A pesar de que con la primera solución el problema planteado estaba casi resuelto, esta segunda solución iba más encaminada a mejorar el aspecto visual y la lectura del mapa por parte del usuario. En esta línea, el resultado final sería un mapa en el que los marcadores quedarían agrupados en conjuntos de marcadores tal y como vemos en este ejemplo:

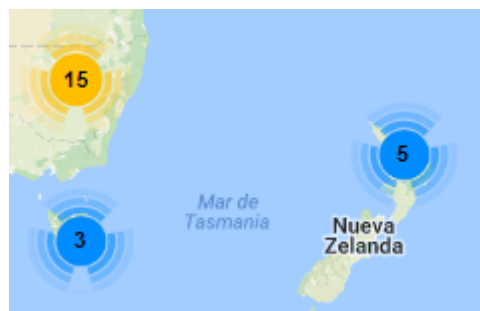


Figura 6.1. Agrupación de marcadores en clústeres

Como vemos en el ejemplo el resultado era lo que se estaba buscando pues el mapa era mucho más legible con este método, además se especifica en cada clúster el número de marcadores que contiene.

6.1.4 Movimiento horizontal en divs

Este problema hace referencia a la fase de diseño del entorno web, más en concreto a la creación de un contenedor situado a la izquierda de la aplicación donde se pudieran visualizar las noticias de cada ubicación.

La idea de este contenedor es que inicialmente cuando hagamos clic en una ubicación se muestren en él de forma resumida los títulos de las noticias y un botón que nos permita el resto de la información. Esa información restante sería mostrada dentro del mismo contenedor, pero en una capa oculta al usuario que solo se mostraría en el caso de querer consultar esa noticia.

El problema surge al intentar implementar ese contenedor de dos capas con movimiento horizontal que permita visualizar/ocultar una de las dos capas. Esto es así porque JQuery no provee ningun método para realizar este tipo de movimientos en horizontal. De igual forma, no se encontró ningun plugin para JQuery que facilitara esta labor dado que con los diferentes plugins probados no se conseguía el efecto deseado.

Tras muchas pruebas se consiguió el movimiento horizontal deseado haciendo uso de animaciones de JQuery que utilizaran el margen izquierdo para realizar los movimientos.

Capítulo 7.

Conclusiones y líneas futuras

En lo referente a las conclusiones que se obtienen del proyecto llevado a cabo, podemos decir que se han satisfecho cada uno de los objetivos inicialmente marcados en el desarrollo del proyecto. Como resultado de la consecución de dichos objetivos se ha creado una aplicación que, aunque puede ser mejorada, es totalmente funcional y satisface las necesidades mínimas de una aplicación web de este tipo.

Por otro lado, el desarrollo de este proyecto ha conllevado el empleo de diferentes herramientas, las cuales en su gran mayoría son bastante usadas actualmente. Este hecho ha supuesto tener un mayor conocimiento de estas tecnologías que, a nivel personal, me abrirá un abanico de posibilidades mayores en el futuro.

También cabe destacar que los conocimientos adquiridos a lo largo de estos años en la carrera han servido de gran ayuda no solo para emplearlos con herramientas ya conocidas, sino también para facilitar la comprensión de otras herramientas desconocidas por mi hasta el momento.

Por último, cabe destacar que una vez finalizada la aplicación web y habiendo alcanzado los objetivos planteados en el desarrollo, se pueden añadir mejoras a la misma consiguiendo una aplicación mucho más completa y con mayor funcionalidad. Algunas de dichas mejoras o ampliaciones podrían ser las que se recogen a continuación:

- Añadir registro de usuarios con redes sociales y que de alguna forma estos pudiesen añadir comentarios a las noticias que consulten
- Implementar la posibilidad de que cada usuario pueda guardar una ubicación o categorías como favorita.
- Implementar que un usuario cualquiera pueda cargar un rss a través de la aplicación y que esta se encargue de realizar el procedimiento y generar el mapa con las noticias ubicadas en él.

- Añadir más categorías de noticias, mejorando con ello las estadísticas y la organización de las noticias.
- Guardar en la base de datos noticias de cada año pudiendo cargar el mapa con noticias de años anteriores
- Mejorar aspecto de la aplicación consiguiendo que sea totalmente responsiva y visible desde cualquier dispositivo.
- Alojamiento de la aplicación web en un sitio web accesible no solo para usuarios de la ULL

Capítulo 8.

Summary and Conclusions

Related to the conclusions drawn from the project, we can affirm that we have met all the goals initially named. As a result of the achievement of these goals it has been created an application that is completely functional and which satisfies the minimum needs of a web application of this type, although it can be improved.

At the same time, to be able to develop this project we have needed the use of different tools, most of them which are widely used today. This fact has meant having a greater knowledge of these technologies that, on a personal level, will open me a wide range of possibilities.

In addition, it should be notice that knowledge acquired over these years during the career have been a great help, not only to use them with tools I have already known, but also to facilitate the learning of another tools which are unknown to me so far.

Finally, it is also important that once the web application is finished and the objectives which are set in the development have been achieved, we can obtain such a much more complete and functional application by adding some improvements. Some of the improvements or extensions could be those listed below:

- Add registration of users with social networks and in some way, these could add comments to the news that they consult.
- Implement the possibility that each user can save a location or category as a favorite.
- Implement that any user can upload a rss through the application and that this is responsible for performing the procedure and generate the map with the news located in it.
- Add more news categories, thereby improving statistics and news organization.

- Save in the database news of each year being able to load the map with news of previous years.
- Improve the appearance of the application making it fully responsive and visible from any device.
- Hosting the web application on a website accessible not only for ULL's users.

Capítulo 9.

Presupuesto

A continuación, se ha realizado un presupuesto estimado de lo que costaría realizar este proyecto y las herramientas a utilizar en él.

El coste de las herramientas en un precio aproximado pues esto puede variar en base a las exigencias de calidad, así como las variaciones habituales que se producen en el mercado de las tecnologías.

9.1 Estimación coste de herramientas

Tipos	Precio/unidad	Total
Servidor Web y FTP	700€	700€
Servidor DNS	600€	600€
Dominio	7€/año	35€/5años
TOTAL		1475 €

Tabla 9.1. Estimación de coste de herramientas.

9.2 Estimación coste de desarrollo e implantación de la aplicación

La siguiente estimación de coste está presupuestada para un programador junior con coste de 10 euros la hora.

Tipos	Horas	Total
Configuración de los servidores	10	100 €
Implantación de la base de datos	8	80 €
Implementación de la aplicación web	168	1680 €
Testeo y depuración	48	480 €
TOTAL	234	2340 €

Tabla 9.2. Estimación de coste de desarrollo.

9.3 Presupuesto final estimado

El presupuesto final del proyecto sería por tanto el coste empleado en la compra de las herramientas necesarias para poner en marcha la aplicación web, así como el coste del desarrollo y puesta en marcha de la aplicación. Teniendo en cuenta las cantidades calculadas anteriormente **el presupuesto final es de 3815 €.**

Apéndice A.

Diagrama de prueba

A continuación, se muestra un diagrama de la prueba de 30 noticias realizadas donde se detalla la noticia, la ubicación obtenida y si la ubicación obtenida es correcta o no.

Título	Descripción	Ubicación obtenida	¿Válida?
Operativo de limpieza en Los Gladiolos y Azorín	Servicios Públicos concluyó un nuevo operativo del programa de zafarranchos de la Operación Barrios. La iniciativa se despliega por varias vías de los barrios de Los Gladiolos y Azorín, en el distrito Salud- La Salle.	LOS GLADIOLOS	Sí , es una ubicación de Santa Cruz de Tenerife y además se encuentra en la BBDD. De no estar "Los Gladiolos" hubiera detectado "Azorín" como ubicación y también sería válida.
Duggi abrirá el programa "Refréscate en tu distrito"	La plaza de Duggi fue la primera parada del ciclo de actividades infantiles organizadas este mes por el distrito Centro-Ifara en barrios del municipio para que los más pequeños puedan disfrutar del verano	DUGGI	Sí
El ascensor del Parque Marítimo ya funciona	En la mañana de ayer entrará en funcionamiento el ascensor instalado en el Parque Marítimo, que permitirá el acceso de los usuarios desde los aparcamientos hasta la planta donde se encuentran las piscinas.	PARQUE MARITIMO	Sí
Unidos por un infarto	Dice el sabio refranero español que "es de bien nacido ser agradecido", y un vecino de Ofra lo ha cumplido al pie de la letra. No era para menos cuando lo que estuvo en juego fue nada más y nada menos que su vida	OFRA	Sí
La reapertura de la iglesia el día 15 abre las fiestas de Las Nieves en Taganana	La reapertura de la iglesia, tras las obras de rehabilitación, abre el próximo día 15 las fiestas patronales de Taganana en honor a Nuestra Señora de las Nieves que se extenderán hasta el 28 de agosto	TAGANANA	Sí
La demolición del edificio de aparcamientos de Las Teresitas costará 540.000 euros	La Junta del Gobierno del Ayuntamiento de Santa Cruz de Tenerife aprobó este lunes el proyecto de demolición total del edificio de aparcamientos situado en la cabecera de la playa de Las Teresitas	LAS TERESITAS	Sí , al igual que en el primer caso son válidas tanto "Las Teresitas" como "Ayuntamiento de Santa Cruz de Tenerife"
Si se puede pedir mejorar de forma urgente la accesibilidad en El Toscal	Para el grupo político es indispensable eliminar tres barreras del barrio santacrucero	TOSCAL	Sí
El Sobradillo critica el mal estado del polideportivo	Vecinos de El Sobradillo denunciaron ayer el mal estado en el que se encuentra la única cancha deportiva del barrio, ubicada entre las calles Sirinoque, Arroyo, Garza e Iris	EL SOBRADILLO	Sí

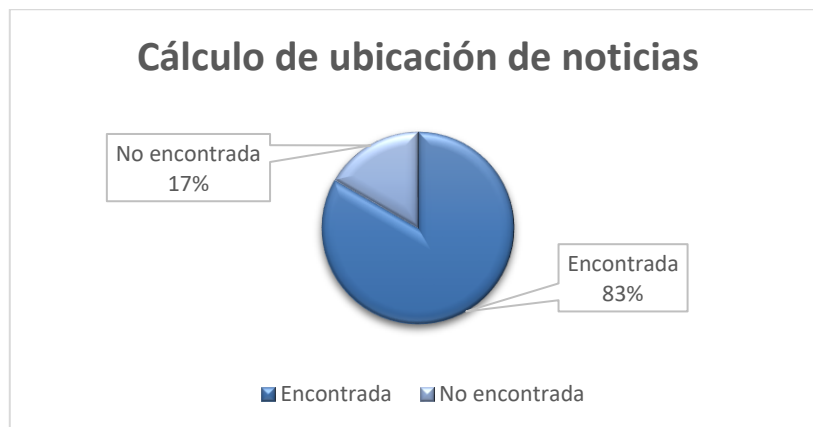
El manga reina en la Summercon	Miles de jóvenes acuden a la feria, que se desarrolla en el Recinto Ferial y el Auditorio y que reúne a dibujantes de Marvel, 'youtubers', escritores y animadores de Hollywood	RECINTO FERIAL	Sí
Cierran el paso de vehículos a Cueva Roja para evitar actos vandálicos	El Ayuntamiento de Santa Cruz de Tenerife ha procedido en días recientes a cerrar el paso de vehículos al interior de Cueva Roja, atendiendo a las peticiones formuladas por los vecinos de la zona, pues se había alertado de actos vandálicos en el interior de la cueva que generaban ruido y malestar	CUEVA ROJA	Sí
Destinan 506.000 euros a la mejora de los pluviales de Acoran	El Cabildo, en colaboración con el Ayuntamiento de Santa Cruz, llevará a cabo la mejora de la red de pluviales en la urbanización Acoran por un importe de más de 506.000 euros	ACORAN	Sí
Las obras de mejora del quiosco de la plaza Weyler acabarán en septiembre	Las obras de remodelación del quiosco de la plaza Weyler dieron comienzo estos días, una vez firmada el acta de replanteo y aprobado el preceptivo plan de seguridad. Los trabajos corren a cargo de la empresa concesionaria, y se prevé que puedan finalizar en septiembre	WEYLER	Sí
Los adoquines de Bravo Murillo serán sustituidos	El Ayuntamiento de Santa Cruz iniciará la próxima semana el reasfaltado que sustituirá a los adoquines de un tramo de la avenida Bravo Murillo.	BRAVO MURILLO	Sí
La Biblioguagua vuelve a prestar sus servicios tras el paréntesis veraniego	En su reencuentro con los vecinos, este vehículo visitará La Salud, Chimisay, Echeyde, Juan XXIII y Las Cabritas entre otros lugares	LA SALUD	Sí
La Comisión Especial de Sugerencias y Reclamaciones ahorrará más de 200.000 euros durante este mandato	El actual presidente de la CESR del Ayuntamiento de Santa Cruz de Tenerife, Enrique Rosales, consigue reducir los gastos y mejorar la visibilidad y efectividad del órgano	AYUNTAMIENTO DE SANTA CRUZ DE TENERIFE	Sí
La Guardia Civil rescata a 50 inmigrantes de 3 pateras en la costa de Almería	La Guardia Civil ha rescatado este jueves a un total de 50 inmigrantes de tres pateras que han sido interceptadas en las proximidades de la costa de Almería	GUARDIA CIVIL	No , hace referencia a la guardia civil de Almería no a la de Santa Cruz
Grave tras caer en unas rocas en Arico	Un varón de 36 años sufrió ayer heridas de carácter grave al caer en una zona de rocas en Porís de Abona, en el Arico, según informó el Centro Coordinador de Emergencias y Seguridad (Cecoes) 1-1-2	CENTRO COORDINADOR DE EMERGENCIAS	No , a pesar de hacer referencia al Centro Coordinador de Emergencias el cual se encuentra en Santa Cruz se trata de una noticia de Arico.
Solicita un proyecto sobre la historia de los barrios	Ciudadanos (Cs) demandó ayer al Ayuntamiento de Santa Cruz de Tenerife que "cree un proyecto sobre la historia de los barrios de la capital"	AYUNTAMIENTO DE SANTA CRUZ DE TENERIFE	Sí
Conato al arder ropa por fuera del albergue municipal	Bomberos del Consorcio de Tenerife destinados en el parque de Tome Cano acudieron en la madrugada de ayer a sofocar un conato en el exterior del albergue de la capital tinerfeña	TOME CANO	Sí

Un vecino de Finca España fallece en un accidente en Los Baldíos	Un motorista de 32 años, natural de La Palma y vecino de Finca España, falleció a primera hora de la tarde de ayer al colisionar con un coche en el municipio de La Laguna	FINCA ESPAÑA	Sí
Recuerdo de su colegio para doña Lolita	Emoción hasta las lágrimas y, sobre todo, recuerdo sentido a quien fuera maestra, amiga o familia. Casi setenta personas se congregaron en el pueblo marinero para asistir al homenaje que la comunidad educativa del CEIP San Andrés le brindó a María Dolores Solinis Llata	AYUNTAMIENTO DE SANTA CRUZ DE TENERIFE	Sí
Taller cuestionado por los vecinos	La Plataforma por Anaga ha respondido con un comunicado de cuatro puntos al anuncio del inicio del proceso de participación de la Reserva de Biosfera por parte del Cabildo que, aseguran, "rechazamos la forma, el fondo y el lugar elegido: la Dársena Pesquera"	ANAGA	Sí
Entre olvido y sabor a caramelo en La Gallega	No quieren dar la cara, pero están hartos de soportar durante años lo que entienden como "desidia y desinterés de unas administraciones que se pasan la pelota.	LA GALLEGA	Sí
IUC ve "dejación municipal" con el Castillo de Paso Alto	Izquierda Unida Canaria (IUC) criticó ayer la "dejación" del Ayuntamiento de Santa Cruz con el Castillo de Paso Alto, una propiedad municipal declarada Bien de Interés Cultural (BIC)"	IUC	Sí , noticia relacionada con IUC a la cual se le atribuye la ubicación de su sede, situada en Santa Cruz
Loro Parque, el mejor zoológico del mundo	El portal de viajes TripAdvisor le otorgó el premio Travellers' Choice 2017, que se asigna en función de las evaluaciones independientes que hacen los usuarios de la plataforma y que han visitado las instalaciones.	No encontrada	Sí , no se hace referencia a ninguna zona de Santa Cruz de Tenerife o La Laguna
Martínez elevará al Consejo Consultivo las extras de la Local si no hay acuerdo	La muestra podrá visitarse hasta el próximo día 21 y está compuesta de 23 leos	No encontrada	Sí , no se hace referencia a ninguna zona de Santa Cruz de Tenerife o La Laguna
Cuando se habla de las citas españolas de videojuegos nosotros siempre estamos ahí	La feria tecnológica propone una jornada para toda la familia, en la que padres e hijos podrán disfrutar hoy de actividades dedicadas al uso de las nuevas tecnologías desde la infancia	No encontrada	Sí , no se hace referencia a ninguna zona de Santa Cruz de Tenerife o La Laguna
El Club de Golf da 3.800 euros a Padre Laraña	Este viernes se escenifica el combate contra las tropas inglesas en la plaza de la Candelaria	CLUB DE GOLF	Sí
Vulcan, el robot capaz de sumergirse en lava	La colaboración policial del Departamento de Seguridad Interior de los Estados Unidos permitido detener a un zamorano investigado por un delito de distribución de pornografía infantil	No encontrada	Sí , no se hace referencia a ninguna zona de Santa Cruz de Tenerife o La Laguna
Las ondas gravitacionales tardan 10 millones de años en producirse	Si dos galaxias colisionan, la fusión de sus agujeros negros centrales desencadena ondas gravitacionales, con una ondulación que recorre el espacio	No encontrada	Sí , no se hace referencia a ninguna zona de Santa Cruz de Tenerife o La Laguna

A.1. Graficas de análisis

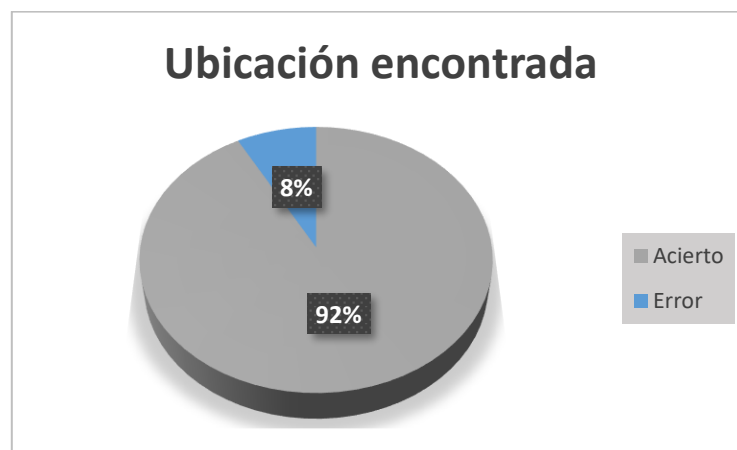
En este apartado se encuentran diferentes gráficas que nos permiten apreciar de una manera más sencilla los resultados de la tabla anterior.

En primer lugar, tenemos un gráfico que muestra el porcentaje de noticias que se han ubicado para la muestra de 30 noticias:



Por otro lado, se han realizado dos gráficos adicionales donde se puede ver el porcentaje de acierto y error en el cálculo de las ubicaciones. Se han analizado los dos casos posibles:

- Ubicación encontrada, donde:
 - El porcentaje de acierto indica la efectividad en el cálculo de las ubicaciones.
 - El porcentaje de error indica el porcentaje de noticias que han sido ubicadas pero que no deberían haberse ubicado por diferentes motivos como los que se reflejan en la tabla del apartado anterior.



- Ubicación no encontrada, donde:
 - El porcentaje de acierto indica que estas noticias no era posible ubicarlas.
 - El porcentaje de error indica que se debería haber encontrado una ubicación para la noticia.



Bibliografía

- [1] Tutorial mongodb, <https://programmerblog.net/php-mongodb-tutorial/>.
- [2] Tutorial fpdf, <https://desarrolloweb.com/articulos/funciones-fpdf.html>.
- [3] FPDF, <http://www.fpdf.org/>.
- [4] Tutorial mongodb en github, <https://github.com/Elolawyn/MongoDB-PHP7-Tutorial>.
- [5] Tutorial mongodb, <https://programmerblog.net/php-mongodb-tutorial/>.
- [6] Apache, <https://httpd.apache.org/>
- [7] PHP, <http://php.net/manual/es/index.php>
- [8] Bootstrap, <http://getbootstrap.com/>
- [9] jQuery, <https://jquery.com/>
- [10] API Google Maps, <https://developers.google.com/maps/?hl=es-419>