



**Universidad**  
Zaragoza

1542

# Trabajo Fin de Grado

## “WikiData plugin para WordPress”

Autor

D. Ezequiel Barbudo Revuelto

Directora

Dra. Dª Piedad Garrido Picazo



**Escuela Universitaria  
Politécnica - Teruel**  
**Universidad Zaragoza**

Escuela Universitaria Politécnica de Teruel

2017



# Resumen

A través de la plataforma colaborativa Phabricator, Wikimedia Foundation administra la gestión de bugs de sus sitios web y las posibles mejoras de software de los mismos, centrada en el desarrollo colaborativo, recibe feedback de posibles problemas encontrados por los usuarios para resolverlos y acepta proposiciones de nuevos proyectos que puedan mejorar o aumentar las funcionalidades de los proyectos de Wikimedia.

Este TFG busca atender, como colaborador de Wikimedia, una de estas peticiones de nuevo proyecto relacionado con Wikimedia, en concreto, de la tarea T138371, consistente en la demanda del desarrollo de un plugin para WordPress que permita asociar etiquetas y categorías de una página Web creada en WordPress con los identificadores de elementos de Wikidata que pudieran tener relación, para posteriormente introducir enlaces y metadatos en la Web. Finalmente, se liberará el código del plugin bajo una licencia GPL 2.0 o posterior.

**Palabras clave:** Plugin, Wikimedia, Wikidata, WordPress, Metadatos

# Abstract

Through the Phabricator collaborative platform, Wikimedia Foundation manages its websites bugs and its possible software improvements, focused on collaborative development. It also gets feedback of problems found by users, solving them and accepting new Project propositions which may improve or increase Wikimedia projects functionalities.

This Final Degree Project pursues, as Wikimedia collaborator, to attend one related to Wikimedia Project proposition, specifically, task nº T138371, which consists in a novel request to develop a plugin for WordPress that allows users to associate tags and categories of a website with Wikidata ID items. Later, links and metadata will be entered on the Web. Finally, the code will be released under a GPL 2.0 or later license.

**Keywords:** Plugin, Wikimedia, Wikidata, WordPress, Metadata



# Tabla de Contenidos

Resumen .....	4
Abstract .....	4
Tabla de Contenidos .....	6
Agradecimientos .....	10
1 Introducción .....	12
2 Objetivo .....	14
3 Estado del arte .....	16
3.1 WordPress .....	16
3.2 ¿Qué es un plugin? .....	16
3.3 Wikidata .....	17
3.4 Etiquetas y categorías .....	17
3.5 Microformatos .....	18
3.6 Desarrollos similares .....	18
4 Etapa de análisis .....	20
4.1 Requisitos .....	20
4.1.1 Requisitos funcionales (RF) .....	20
4.1.2 Requisitos NO funcionales (RNF) .....	20
4.2 Sistema Operativo (SO) .....	21
4.3 Lenguajes de programación .....	21
4.4 El ciclo de carga de WordPress .....	21
4.5 La API de Wikidata .....	23
4.5.1 Estructura de la API de Wikimedia .....	25
4.5.2 Llamada a la API de Wikidata .....	26
4.6 Metodología y plan de trabajo .....	26
4.6.1 Fase 1: concepción .....	28
4.6.2 Fase 2: elaboración .....	28
4.6.3 Fase 3: construcción .....	28
4.6.4 Fase 4: transición .....	29
5 Etapa de Diseño .....	30
5.1 Diagrama de casos de uso .....	30
5.2 Diagrama de secuencia .....	31

5.3 Interfaz .....	34
5.4 Base de datos .....	34
6 Etapa de Implementación .....	36
6.1 Tiempo de desarrollo .....	36
6.2 Proceso de implementación.....	37
6.2.1 Buenas Prácticas.....	38
6.2.2 Recuperación y muestra de términos en los ajustes del plugin.....	39
6.2.3 Recuperación de información de Wikipedia .....	39
6.2.4 Cambio de paradigma .....	41
6.3 Estado final.....	41
7 Conclusiones y trabajo futuro .....	42
8 Licencia software y documental.....	44
9 Referencias bibliográficas.....	46
10 Glosario de términos .....	48
Anexo I: Manual de Instalación y uso de Wikidata References.....	52
Requisitos previos a la instalación .....	52
Opciones de instalación .....	52
Manual de uso.....	53
1 Términos, etiquetas y categorías .....	53
2 Creación de nuevos términos para etiquetas o categorías.....	54
3 Edición de términos.....	55
4 Asociación de términos con Wikidata .....	56
5 Muestra de información sobre asociaciones .....	57
6 Ajustes .....	58
7 Cambios del plugin sobre la Web según los ajustes.....	60
8 Desinstalación .....	64
Anexo II: La veracidad de los datos de Wikidata .....	66
1 Principios de desarrollo de artículos .....	67
2 La calidad de la información.....	69
Calidad intrínseca .....	69
Calidad contextual.....	69
Calidad representacional.....	70
Calidad del acceso a la información .....	70
3 La calidad de la información digital.....	71

Bibliografía .....	72
--------------------	----



# Agradecimientos

En primer lugar, quisiera agradecer al equipo de colaboradores de Wikimedia Foundation: Andy Mabbett, David Abián, Lydia Pintscher, Jesús Tramullas, Dan Brickley y Philip John por su ayuda, y muy especialmente a mi directora, Piedad Garrido por depositar su confianza en mi para desarrollar este proyecto.

Quisiera también mencionar, al equipo de profesores de la Escuela Universitaria Politécnica de Teruel y cómo no, a mis compañeros de carrera, algunos de los cuales yauento como amigos para toda la vida.

Finalmente, a mi familia, por los valores que me han transmitido desde siempre, a mi pareja, por su apoyo incondicional y muy especialmente, a mi padre, ejemplo de trabajo y constancia para cualquiera que lo conozca.

First, I would like to thank the Wikimedia Foundation collaborators team: Andy Mabbett, David Abián, Lydia Pintscher, Jesús Tramullas, Dan Brickley and Philip John their help and especially to my project director, Piedad Garrido, for trusting me to develop this project.

I also would like to mention the team of teachers of the Escuela Universitaria Politécnica de Teruel and, of course, my career partners, some of which I tell as friends for life.

Finally, to my family, for the values they have always instilled to me, to my twosome, for her unconditional support, and very specially to my father, example of work and perseverance for anyone who knows him.



# 1 Introducción

Wikimedia Foundation es una organización sin ánimo de lucro dedicada al desarrollo y distribución de contenido educativo de carácter gratuito y multilingüe, con el objetivo de conseguir un mundo en el que sea posible acceder a todo el conocimiento generado por la humanidad, de forma libre, a través de varios proyectos.

Debido a su carácter no lucrativo, el desarrollo de la misma es llevado a cabo por voluntarios, tanto a nivel de contenido, como de software. En el caso de este último, Wikimedia hace uso de la suite Phabricator para la gestión colaborativa del desarrollo de su software, donde se pueden reportar errores, sugerir mejoras o pedir nuevas funcionalidades.

El contenido de esta memoria se basa en la atención de una de estas peticiones de nuevas funcionalidades relacionadas con el proyecto de Wikimedia, en este caso, la petición T138371<sup>1</sup> en la que se requiere un plugin para la herramienta de gestión de contenidos WordPress que sea capaz de relacionar los términos utilizados como etiquetas en la misma, con elementos de Wikidata, proyecto de Wikimedia destinado a la colección de datos estructurados como base de datos secundaria.

Esta petición supone asociar las etiquetas utilizadas en el blog de WordPress con elementos de Wikidata con el fin de añadir, en las páginas en las que aparezcan esas etiquetas, los metadatos correspondientes a esa asociación, así como posiblemente, enlaces directos a los elementos de Wikidata en la página, mediante un plugin para WordPress.

Estos metadatos, entendidos como información que describe y caracteriza contenido, condiciones, autoría o procedencia de los datos (entre otros), permite a los motores de búsqueda realizar aproximaciones más exactas en relación a los términos de búsqueda de los usuarios, y junto con otras variables, son una herramienta muy útil para el posicionamiento Web (SEO).

Así, este proyecto supone el estudio de las tecnologías y APIs de WordPress y Wikidata, y el posterior desarrollo del citado plugin, a fin de satisfacer las peticiones de la tarea de la plataforma Phabricator de Wikimedia, trabajando conjuntamente con el usuario que la inició, con el objetivo de que el mismo cumpla con sus necesidades.

Finalmente se liberará dicho plugin bajo una licencia de código libre GPL 2.0<sup>2</sup> o posterior, de modo que sea posible seguir manteniendo o mejorando el plugin con la ayuda de la comunidad de Wikimedia.

---

<sup>1</sup> "Wordpress plugin to associate tags with Wikidata Ids": <https://phabricator.wikimedia.org/T138371>

<sup>2</sup> General Public License, version 2: <https://www.gnu.org/licenses/old-licenses/gpl-2.0.en.html>



## 2 Objetivo

El objetivo principal de este proyecto consiste en atender la petición de creación de un plugin para WordPress que asocie las etiquetas utilizadas en la Web gestionada por el mismo, con el identificador (en adelante id) de términos específicos de Wikidata. Esta petición, creada el 22 de junio de 2016 por [Andy Mabbet](#)<sup>3</sup> en la plataforma Phabricator de Wikimedia, se encuentra disponible en la tarea “WordPress plugin to associate tags with Wikidata IDs”.

Las funcionalidades de este plugin consisten en la asociación de términos, y la inclusión de ciertos metadatos en la Web usando un formato preestablecido. Para ello, la primera tarea de este Trabajo Final de Grado (TFG), consiste en conocer y comprender el funcionamiento del sistema de gestión de contenidos WordPress y de la API (Application Programming Interface) de Wikidata. Para así en segunda instancia, proceder a la implementación del plugin, siguiendo las guías de diseño e implementación de WordPress, haciendo uso de la API de Wikidata, acabando con un producto que satisfaga las necesidades de la tarea de la que depende el mismo.

Finalmente, se publicará el plugin bajo una licencia GPL, en la tienda de plugins de WordPress, para que la instalación del mismo sea lo más sencilla, accesible y visible para los usuarios.

---

<sup>3</sup> Andy Mabbett: <https://commons.wikimedia.org/wiki/User:Pigsonthewing>



# 3 Estado del arte

En los últimos años, las tecnologías de desarrollo respecto al diseño y desarrollo Web han experimentado cambios radicales en cuanto a su evolución, la llegada al mercado de plataformas gestoras de contenido ha puesto al alcance de los usuarios la creación de nuevos portales Web sin la necesidad de escribir una línea de código, y facilitado, a desarrolladores y administradores, la concepción de nuevas funcionalidades más complejas que se adapten a las nuevas necesidades del mercado.

Pese a que existen una gran variedad de sistemas gestión de contenidos, el proyecto se basa en WordPress y es sobre el que nos centraremos. A continuación, se va a proceder a comentar los elementos necesarios para la correcta implementación del plugin solicitado, para finalizar concluyendo que no existe ningún desarrollo similar al respecto.

## 3.1 WordPress

WordPress es un sistema de gestión de contenidos, de código abierto, que fue originalmente enfocado a la creación de blogs, pero que con los años y la deriva del mercado, ha evolucionado hasta ser capaz de crear cualquier tipo de sitio Web, incluso dedicados al comercio electrónico. Desarrollado en PHP, para entornos MySQL y Apache, bajo una licencia GPL, es en la actualidad usado por el 29% de los sitios de Internet [10] y por el 60% de los sitios Web [8] creados con un CMS (Content Management System).

Al liberar su código, es posible modificar el mismo para adaptarlo a las necesidades de cada caso, no obstante, pese a que esto es una opción, el punto fuerte de WordPress es su capacidad para admitir extensiones, gracias a la forma en la que funciona. Así, en lugar de cambiar el código, el núcleo del software (en inglés core) se mantiene mediante las actualizaciones oficiales, y las funcionalidades extra se añaden a través de plugins o plantillas, convirtiéndose en un CMS potente, flexible y escalable, siendo a su vez muy sencillo de instalar y configurar.

## 3.2 ¿Qué es un plugin?

En términos de desarrollo de software, un plugin es un software o aplicación que se acopla a otra mayor para extender sus funcionalidades, el plugin es ejecutado por la aplicación principal y ambos se comunican entre sí mediante una API.

Un plugin no supone un parche o una actualización de la aplicación a la que se adhiere, sino que su objetivo es añadir nuevas funcionalidades que satisfagan necesidades, por lo general, bastante específicas. Además, una de las principales ventajas de estos complementos, es que no se toca el núcleo de la aplicación, por lo que el mantenimiento o posibles errores provocados por la extensión, se reducen.

### 3.3 Wikidata

Wikidata es una base de datos secundaria libre y colaborativa, que recopila datos estructurados para dar soporte a Wikipedia, Wikimedia Commons, otras Wikis del proyecto Wikimedia y a cualquier persona del mundo actuando como almacenamiento de datos central de las Wikis. El contenido de Wikidata está disponible bajo una licencia Creative Commons (CC), puede ser exportado usando formatos estándar como JSON (JavaScript Object Notation) o XML (eXtensible MarkUp Language) y puede ser interconectada con otros proyectos de almacenamiento de información libres.

El propósito de Wikidata es crear una base de datos de dominio público, de hecho, los datos de Wikidata son publicados bajo una licencia CC 1.0, lo que permite el uso de los mismos en distintos escenarios, es posible copiar, modificar, distribuir y presentar la información, incluso con propósitos comerciales.

Wikidata no solo actúa como base de datos al uso [1], sino como base de datos secundaria, recopilando las fuentes de información, reflejando el origen de las mismas y apoyando su veracidad. El desarrollo e introducción de los datos se realiza de forma colaborativa, pudiendo ser introducidos y revisados por editores de Wikidata, o a través de robots (en inglés bots), traduciéndose automáticamente a cualquier idioma y, fomentando así la corrección y edición de las traducciones entre los editores. Toda esta información se recoge de forma estructurada, lo que permite que pueda ser fácilmente reutilizable por otros proyectos de Wikimedia, por terceros, y/o ayudando al desarrollo de proyectos similares de procesado de información.

Esta forma plurilingüe de recolectar datos estructurados sirve como apoyo a Wikimedia, proporcionándole archivos de datos interlingüísticos, fácilmente mantenibles, lo que reduce el trabajo de mantenimiento de las Wikis mejorando su calidad, y cediendo esta información a terceros, que pueden acceder de diferentes maneras.

### 3.4 Etiquetas y categorías

A la hora de crear una entrada en WordPress, existe la posibilidad de añadirle estos dos conceptos: (i) etiquetas y (ii) categorías que permiten clasificar las entradas mediante términos o conceptos. Ambas forman parte, en WordPress, del concepto de taxonomía. Una taxonomía es una clasificación concreta de esos términos y conceptos, ayudando a clasificar y organizar las entradas de su sitio Web, aumentando así la usabilidad para el usuario, siendo capaz de buscar en el sitio Web no sólo de forma cronológica sino a través de las temáticas que crean las taxonomías. La diferencia radica en que las categorías forman una estructura jerárquica del contenido, siendo las etiquetas una clasificación más liviana.

Ambas ayudan a la navegación interna del sitio WordPress, y como si de términos para SEO (Search Engine Optimization) se tratara, la inclusión de éstas en una entrada debe ser selectiva, ya que marcará la diferencia a la hora de la navegación por temáticas del sitio Web. Cada taxonomía está formada por varios atributos comunes:

- **Identificador:** transparente para el usuario, para uso de WordPress
- **Nombre:** término o texto identificativo de la taxonomía
- **Slug** (sin traducción): término identificativo de la taxonomía saneado, es decir, sin caracteres especiales o mayúsculas

Además, pueden tener otros atributos como una descripción, u otros términos asociados. Por ejemplo, el caso de las categorías para generar una jerarquía e incluso es posible crear nuevas taxonomías con atributos propios.

### 3.5 Microformatos

Los microformatos son una forma simple de añadir significado semántico a un contenido legible por un humano, formando una estructura y jerarquía que los convierte en un lenguaje de marcado que permite introducir contenido en los recursos de información usados por algunas etiquetas HTML (HyperText Markup Language). Para que un microformato tenga validez, los valores de los elementos a añadir deben estar definidos, estructurados y jerarquizados.

Una de las peticiones para el plugin es añadir estos microformatos a los enlaces de las etiquetas, incluyendo las URL (Uniform Resource Locator) de los términos de Wikidata asociados con la estructura de microformatos Schema.org (<http://schema.org>) fundada por Google, Microsoft, Yahoo y Yandex, dedicada a crear, mantener y promover esquemas para datos estructurados en Internet.

### 3.6 Desarrollos similares

En la actualidad no existe ningún plugin para WordPress que automatice la asociación de las etiquetas o categorías creadas en WordPress con un término de Wikidata. Por un lado, sí que existen, no obstante, variedad de plugins que permiten añadir metadatos a la Web con el objetivo de mejorar el SEO, sin embargo, tampoco es posible añadir con estas extensiones metadatos precisos de Wikidata a páginas concretas de la misma.

Por otro lado, el único plugin que añade microformatos a ciertos elementos de una Web WordPress, conocido como “All in one Schema Rich Snippets”<sup>4</sup> que soporta microformatos para reseñas (en inglés reviews), eventos, usuarios, productos, recetas, software, vídeo y artículos, por lo que no puede ser usado como punto de partida puesto que su objetivo principal, es el de añadir fragmentos enriquecidos (en inglés rich snippets) al contenido de los posts de WordPress y por supuesto, no permite ningún tipo de asociación con elementos externos, siendo el usuario el que rellena los campos predeterminados para un conjunto de valores muy concretos, en ningún caso existiendo la posibilidad de asociar alguno de estos fragmentos con otro elemento de WordPress que no sea un Post, no cumpliendo así con las necesidades que debe cubrir el plugin a desarrollar.

---

<sup>4</sup> All in one Schema Rich Snippets, WordPress plugin, <https://es.wordpress.org/plugins/all-in-one-schemaorg-rich-snippets/> [último acceso: 16 de noviembre de 2017]

Por tanto, la iniciativa de análisis, diseño e implementación de un plugin con esta funcionalidad, está justificada y es novedosa, existiendo la necesidad de crear asociaciones efectivas entre términos de los sitios Web con términos de Wikidata y la inclusión de metadatos relacionados a esa asociación, así como la ausencia de cualquier otra extensión para WordPress que sea capaz de satisfacerla.

# 4 Etapa de análisis

Llegados a este punto y siguiendo las peticiones realizadas en la tarea del Phabricator de Wikimedia, se requiere de un plugin que sea capaz de:

- Asociar etiquetas y categorías con términos de Wikidata que estén relacionados
- Permitir al usuario elegir qué término de Wikidata va a asociar a cada etiqueta o categoría, en adelante, a esta acción nos referiremos como “desambiguación”
- Incluir un enlace al término de Wikidata, desde las páginas de archivo de etiquetas y categorías
- Añadir, en las páginas de archivo de etiquetas y categorías, metadatos que asocie el enlace al término de Wikidata como enlace alternativo
- Colocar los enlaces a las páginas de archivo, en un microformato que añada microdatos del enlace con la URL del término de Wikidata

## 4.1 Requisitos

### 4.1.1 Requisitos funcionales (RF)

- RF-1: El plugin deberá integrarse en WordPress y tomar información sobre las etiquetas y las categorías
- RF-2: El plugin deberá conectarse a la API de Wikidata y hacer una búsqueda por término con el “slug” de la etiqueta o categoría
- RF-3: El plugin deberá mostrar una interfaz que permita al usuario desambiguar el término que está eligiendo
- RF-4: El plugin deberá guardar la información del término de Wikidata elegido en WordPress
- RF-5: El plugin deberá incluir un enlace al término de Wikidata asociado a cada etiqueta o categoría en las correspondientes páginas de archivo
- RF-6: El plugin deberá añadir metadatos con la información del término de Wikidata asociado a cada etiqueta o categoría en las correspondientes páginas de archivo
- RF-7: El plugin incluirá en una estructura de microdatos los enlaces a páginas de archivo de etiquetas o categoría, así como un enlace al término de Wikidata asociado a la etiqueta o categoría a la que pertenece el enlace
- RF-8: Si una etiqueta o categoría tiene su campo de asociación con un elemento de Wikidata vacío, se eliminarán los datos añadidos y no se añadirá ningún tipo de metadatos relacionado
- RF-9: El usuario debe ser capaz de elegir entre las opciones del plugin, existiendo la posibilidad de desactivar sus funciones

### 4.1.2 Requisitos NO funcionales (RNF)

- RNF-1: El código del plugin seguirá las guías de desarrollo de WordPress
- RNF-2: El plugin deberá integrarse dentro del propio funcionamiento de WordPress, siendo su inclusión lo menos llamativa posible
- RNF-3: La instalación del plugin deberá generar el mínimo impacto posible en el rendimiento de WordPress

- RNF-4: El funcionamiento del plugin debe ser lo más independiente posible del usuario, no necesitando de ajustes o mantenimiento extra, más allá de la asociación de términos y la elección de datos a añadir a su sitio Web.
- RNF-5: La desinstalación del plugin de la aplicación a la que se adhiere debe ser lo más aséptica posible, no dejando datos residuales en la Web o la base de datos

## 4.2 Sistema Operativo (SO)

En el caso de este proyecto, el Sistema Operativo elegido para el desarrollo es indiferente, así que, por comodidad, y por la facilidad de acceso, se realizará sobre el SO Windows.

## 4.3 Lenguajes de programación

WordPress está desarrollado en PHP Hypertext Preprocessor, HTML y JavaScript, por lo que el desarrollo de las extensiones para WordPress, sean en forma de plugins o plantillas, está sujeto a estos lenguajes.

## 4.4 El ciclo de carga de WordPress

Para iniciarnos en el desarrollo de WordPress es fundamental comprender el ciclo de carga [7] de este software, es decir, su flujo de ejecución. Y para familiarizarnos con esto necesitamos entender el concepto de gancho (en inglés hook). Internamente, WordPress está orientada a eventos, prácticamente todo lo que sucede en WordPress crea un evento. Por ejemplo, acciones para crear una entrada, editarla o guardarla, disparan eventos, y cada evento genera un gancho.

Los ganchos son disparadores que permiten a las funciones de WordPress relacionarse entre sí, y con el resto de extensiones añadidas, prácticamente cada evento en WordPress activa uno de estos ganchos, a los que están anclados otras funciones que se ejecutarán en el momento en el que ocurra el evento asociado. Y es aquí donde viene la potencia de WordPress para admitir extensiones, podemos crear ganchos personalizados y anclar a ellos nuestras funciones.

Existen dos tipos de ganchos [13]:

- Una **acción** va a ser una función que se ejecuta cuando ocurre uno de los eventos, cambiará variables internas, tomará datos, enviará información etc... Permitiéndonos añadir funcionalidades extra en un punto determinado del proceso de carga
- Un **filtro** es una función que modifica el contenido de WordPress antes de mostrarlo o guardarla en la base de datos, interceptando y modificando el comportamiento de la carga y los datos procesados

Gracias a estos elementos, podemos definir una serie de funciones que se ejecuten exactamente cuando sea necesario aplicar las funcionalidades de la extensión a desarrollar.

Para anclar funciones propias a estos eventos, la API de WordPress nos proporciona dos:

```
add_action( string $tag, callable $function, int $priority = 10, int $args = 1)  
add_filter( string $tag, callable $function, int $priority = 10, int $args = 1)
```

- **\$tag:** (requerido) el nombre del gancho al que queremos anclar nuestra función
- **\$function:** (requerido) el nombre de la función que queremos ejecutar en el instante que se genere el evento marcado por el gancho
- **\$priority:** (opcional) prioridad de ejecución respecto a las demás funciones ancladas al evento, mayor cuanto más cercano al cero
- **\$args:** (opcional) número de argumentos que la función acepta

De este modo, la API de WordPress facilita mucho la integración de funciones externas, y por tanto el desarrollo de nuevos plugins y plantillas que amplíen las características y funcionalidades que ofrece por defecto, sin tocar el núcleo del código del CMS.

El flujo de ejecución de WordPress siempre mantiene el mismo orden, disparando los mismos eventos en un ciclo llamado bucle (en inglés loop) o proceso principal de WordPress. En cada iteración, WordPress ejecuta las mismas acciones, a las que podemos añadir las nuestras propias para personalizar al máximo nuestro sitio Web.

Lo primero que hace WordPress es comprobar que los archivos necesarios para la correcta carga de su sistema estén disponibles. Tras ello, conecta con la base de datos para tomar los valores de los ajustes definidos por el administrador, usados para elegir qué archivos recuperar de la base de datos, o devolver un número establecido por defecto.

Una vez decidido, conecta a la base de datos para retornar la información especificada y, almacena los resultados en una variable a la que accederá el bucle utilizando sus valores para mostrarlos en sus plantillas.

El siguiente código es el mínimo necesario para mostrar una página Index totalmente funcional que muestre el contenido de cada artículo:

```
<?php  
    get_header();  
    if ( have_posts() ):  
        while ( have_posts() ):  
            the_post();  
            the_content();  
        endwhile;  
    endif;  
    get_sidebar();  
    get_footer();  
?>
```

Esto nos da un ejemplo de lo sencillo que es crear nuevas plantillas o funcionalidades con WordPress, y lo fácil que es acceder mediante código a las variables del bucle. Por supuesto, cada plantilla tiene su propio bucle, que accederá y mostrará distintos tipos de contenido.

Cada una de las funciones llamadas en el bucle, retorna unos resultados diferentes, y, además, y aquí interviene el paradigma de orientación a eventos, dispara el evento específico para la acción que desempeña, ya sea aplicando acciones o filtros.

Para desencadenar estos eventos hace uso de dos funciones:

```
do_action( string $tag, $args = '' )  
apply_filters(string $tag, mixed $value)
```

La función `do_action` propagará el evento y todas las funciones de acción ancladas al bucle `$tag`, se ejecutarán, al igual que ocurrirá con `apply_filters` y todos los filtros.

## 4.5 La API de Wikidata

Wikidata actúa de repositorio, es decir, de almacenamiento principal de los datos para que el resto de las wikis conectadas puedan acceder a ellos, manteniendo Wikidata, no es necesario que las wikis conectadas realicen un mantenimiento específico e individual de la información, pues el contenido se cargará dinámicamente desde Wikidata.

El repositorio de Wikidata se organiza en elementos, cada uno definido por:

- **Etiqueta:** Nombre del elemento, convenientemente descriptivo, siendo el nombre más común por el que el ítem podría ser conocido no siendo necesariamente único
- **Identificador:** Código identificador único del elemento
- **Descripción:** Frase corta con el propósito de desambiguar el término, aclarando el contexto o tipo de elemento real al que hace referencia el artículo, por ejemplo:

<b>Etiqueta:</b>	Roma	Roma
<b>Descripción:</b>	Ciudad	Película

*Tabla 1 Ejemplo de organización de elementos*

- **Uno o más alias:** Los alias son nombres alternativos opcionales para los elementos que pueden ser buscados del mismo modo que las etiquetas, y como ellas, son específicos de cada idioma, pero puede haber tantos como sean necesarios para el elemento
- **Declaraciones:** Las declaraciones describen características de un elemento de forma detallada, consisten de una **propiedad** y un **valor**. Cada propiedad es identificada con un código numérico precedido por una `P`. Por ejemplo, para una ciudad o un edificio, podemos asignar sus coordenadas geográficas especificando valores de longitud y latitud. Toda esta información

puede ser mostrada en cualquier idioma, en el momento de acceder a los valores, los clientes de las wikis mostrarán los datos más actualizados en su idioma. De esta forma, las declaraciones suponen una asociación semántica entre varios elementos de la propia base de datos creando relaciones entre ellas

Ítem	Propiedad	Ítem
<b>Q42</b>	P69	Q691283
<b>Douglas Adams</b>	Educado en	Instituto St. John's

*Tabla 2 Ejemplo de asociación semántica*

- **Enlaces (en inglés sitelinks):** son enlaces especiales que contienen un sitio y un título, y presentan desde elementos individuales de Wikidata a páginas externas como Wikipedia. Se utilizan tanto para la identificación de un elemento de un sitio externo como para el reemplazo de los enlaces interwiki (que son enlaces a páginas o secciones de páginas de las distintas wikis)

Hay diferentes formas de acceder a los datos de Wikidata: (i) herramientas incorporadas en el propio repositorio, (ii) herramientas externas, o (iii) interfaces de programación (APIs):

Cada elemento o propiedad tiene un URI (Uniform Resource Identifier) permanente que se obtiene al añadir su identificador al espacio de nombres de conceptos de Wikidata: <http://www.wikidata.org/entity/>.

Por ejemplo, el URI del elemento “Wikipedia” en el repositorio de Wikidata es <http://www.wikidata.org/entity/Q8449>. Es posible usar el concepto URI para acceder a información sobre Wikidata tanto a través de un cliente HTTP (HyperText Transfer Protocol) que lo dirigirá a la página de Wikidata con la información del elemento “Wikipedia”, como a través de la API que proporciona.

Para casos en los que no es conveniente el uso de la negociación de contenido (HTTP) por cualquier motivo, también es posible acceder a estos datos en un formato específico añadiendo a la URL el sufijo que indique el formato deseado, como .json, .rdf, .ttl o .nt. Por ejemplo: <http://wikidata.org/entity/Q8449.json> devolverá el elemento Q8449 en formato JSON.

La API de acción de MediaWiki provee de acceso de alto nivel a la información contenida en las bases de datos de MediaWiki, un cliente puede conectarse a una wiki, conseguir información, y publicar cambios automáticamente realizando peticiones HTTP al servicio Web. Por ejemplo, si quisieramos recuperar el contenido de la página principal de Wikipedia en formato JSON, la URL resultante para realizar la petición quedaría como sigue:

```
https://en.wikipedia.org/w/api.php?action=query&titles>Main%20Page&prop=revisions&rvprop=content&format=json
```

#### 4.5.1 Estructura de la API de Wikimedia

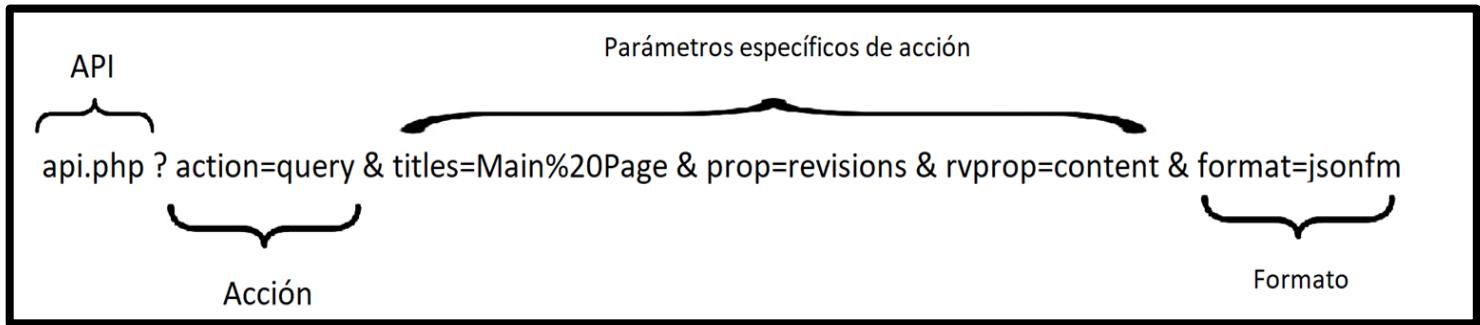


Figura 1 Estructura de una llamada a la API de Wikidata

1. **La API [11]:** el *endpoint* de las wikis suele ser "/w/api.php", por ejemplo, el de Wikidata es: <https://en.wikidata.org/w/api.php>, el *endpoint* es la "página principal" de la API, la URL base para todas las peticiones a la misma
2. **La acción:** El servicio Web de la API de MediaWiki implementa diferentes acciones (todas definidas en las direcciones: "[http://url\\_base\\_de\\_la\\_wiki/w/api.php](http://url_base_de_la_wiki/w/api.php)". En este caso, la acción es "query" para realizar una consulta
3. **Parámetros específicos de acción:** estos son parámetros usados por la acción "query"
  - a. **titles=Main%20Page:** Aquí, configuramos la petición con el contenido que queremos de la página de la Wiki, llamado "Main Page". Si queremos realizar consultas de varias páginas, podemos hacerlo en una sola petición: (titles=pagina1|pagina2|pagina3)
  - b. **prop=revisions:** Podemos hacer peticiones de información o propiedades de una página, este parámetro informa al servicio web que queremos una revisión en concreto de la información de la página. Si no especificamos (como en este caso) la revisión que queremos, nos devolverá la última
  - c. **rvprop=content:** Este parámetro configura la petición para que se nos devuelva la última revisión de la página. Si configuramos un "rvprop=content|user", se nos devolvería la última página de contenido y el nombre del usuario que realizó la revisión más reciente
4. **El formato:** el parámetro "format" permite definir el tipo de formato que queremos que nos devuelva la petición, así, "format=json" devolverá un JSON
  - a. **Acceso a volcados de datos:** Wikidata ofrece copias del contenido disponible para su descarga en varios tipos de datos diferentes. Es necesario tener en cuenta que los volcados de datos en los formatos JSON y RDF (Resource Definition Language) son considerados como interfaces estables pero los de formato XML no
  - b. Los volcados **JSON** con todas las entidades de Wikidata en un solo JSON se pueden encontrar en <https://dumps.wikimedia.org/wikidatawiki/entities/>. Las entidades del JSON no se encuentran necesariamente en ningún orden y son creados semanalmente. En él, cada elemento ocupa una línea del documento, de forma que puede ser leído línea por línea, constituyendo cada una, un objeto JSON individual
  - c. Los volcados **RDF**, pueden ser encontrados, en formato Turtle en <https://dumps.wikimedia.org/wikidatawiki/entities/>, o bien los llamados "fiables" en formato nt
  - d. Pese a que es recomendable usar uno de los formatos anteriores, los volcados **XML** pueden descargarse desde <http://dumps.wikimedia.org/wikidatawiki/>. No obstante, el

formato del JSON dentro del XML es susceptible de cambio sin previo aviso y es posible que sea inconsistente entre revisiones

#### 4.5.2 Llamada a la API de Wikidata

En nuestro caso necesitamos hacer una búsqueda por nombre o término, que serían los nombres de las etiquetas o categorías, que nos devuelva información sobre el identificador del elemento y una descripción del mismo, por lo que haremos uso de la acción “wbsearchentities”.

Siguiendo la estructura de las llamadas a las APIs de Wikimedia, la llamada a realizar para nuestras búsquedas por término quedaría de la siguiente manera:

```
api.php?action=wbsearchentities&origin=&search=termino_a_buscar&format=json&language=idioma
```

Figura 2 Llamada a la API de Wikidata

Esta llamada a la API de Wikidata nos devolverá un JSON con la información de todos los elementos de Wikidata cuyo título coincide con el término que buscamos.

#### 4.6 Metodología y plan de trabajo

A continuación, se dará una explicación de la metodología usada en la planificación seguida para el desarrollo de este TFG, la metodología OpenUp. OpenUP es un proceso interactivo de desarrollo software simplificado, completo y extensible. Es un proceso para pequeños equipos de desarrollo que valoran los beneficios de la colaboración y de los involucrados con el resultado del proyecto, por encima de formalidades innecesarias. Proceso mínimo y suficiente (sólo el contenido fundamental y necesario es incluido). La mayoría de los elementos de OpenUp están declarados para fomentar el intercambio de información entre los equipos de desarrollo y mantener un entendimiento compartido del proyecto, sus objetivos, alcance y avances.

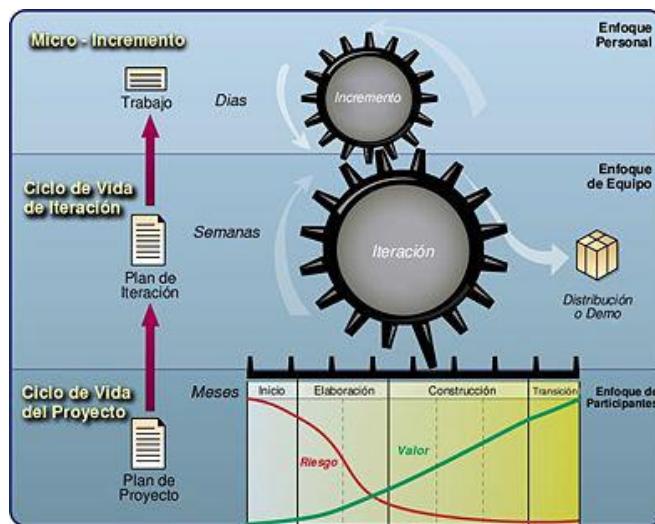


Figura 3 Las tres capas del proceso de desarrollo software con OpenUp

OpenUp posee una serie de principios tales como colaborar para sincronizar y compartir conocimiento, equilibrar las prioridades para maximizar el beneficio obtenido por los interesados en el proyecto, centrarse en la arquitectura de manera temprana para minimizar el riesgo y organizar el desarrollo, otro principio que presenta es el desarrollo evolutivo para obtener retroalimentación y mejoramiento continuo. OpenUp está organizado en dos dimensiones diferentes interrelacionadas entre sí: (i) el método y (ii) el proceso.

- **Método.** El contenido del método es donde los elementos (roles, tareas, artefactos y lineamientos) son definidos, sin tener en cuenta como son utilizados en el ciclo de vida del proyecto
- **Proceso.** Es donde los elementos del método son aplicados de forma ordenada en el tiempo. Muchos ciclos de vida para diferentes proyectos pueden ser creados a partir del mismo conjunto de elementos del método

Los elementos de esta metodología dividen la organización del trabajo en tres niveles: personal, de equipo y de interesados. A nivel personal, los integrantes de un proyecto contribuyen con su trabajo con pequeños incrementos en funcionalidad, denominados microincrementos, los cuales representan los resultados obtenidos en pocas horas o pocos días de trabajo.

OpenUp tiene elementos que ayudan a los equipos de trabajo a enfocar los esfuerzos a través del ciclo de vida de cada iteración, de tal forma que se puedan distribuir funcionalidades incrementales de una manera predecible, una versión totalmente probada y funcional al final de cada iteración.

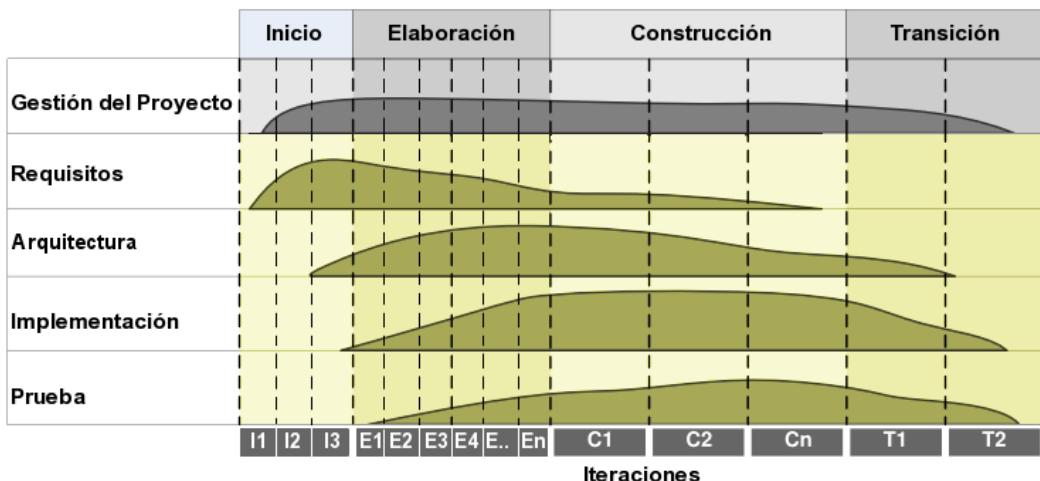


Figura 4 Ciclo de vida de un proyecto según OpenUp

Tal y como se puede observar en la figura 4, el ciclo de vida de un proyecto se estructura en cuatro fases: la concepción, la elaboración, la construcción y la transición. El ciclo de vida del proyecto provee a los interesados de un mecanismo de supervisión y dirección para controlar los fundamentos del proyecto, su ámbito, la exposición a los riesgos, el aumento de valor y otros aspectos.

Para finalizar, comentar que esta metodología es un proceso completo, flexible y corto, fomenta el uso de técnicas ágiles, se puede acoplar a la mayoría de proyectos y tiene un ciclo de vida estructurado que favorece obtener software de calidad. A continuación, se van a explicar con un poco más de detalles cada una de las fases.

#### 4.6.1 Fase 1: concepción

Esta primera fase está dirigida a la selección de una estructura para el plugin y la creación de un prototipo base sobre el que sea posible desarrollar las funciones de la aplicación deseada y la prueba de las llamadas a la API de Wikidata, así como de una fase de pruebas para una total comprensión del funcionamiento de WordPress y su API [4].

En esta fase se realizarán:

- Selección de un framework adecuado para el desarrollo del plugin
- Prueba de la llamada a la API de Wikidata
- Pruebas de desarrollo en WordPress con el objetivo de comprender su paradigma de orientación a eventos y cómo funcionan internamente los ganchos
- Desarrollo de un prototipo de plugin para WordPress de baja fidelidad que sea capaz de realizar pequeñas acciones en determinados eventos [3]

#### 4.6.2 Fase 2: elaboración

Esta segunda fase consistirá en desarrollar sobre nuestro prototipo, una página de ajustes que nos permita asociar las etiquetas de WordPress con elementos de Wikidata, lo cual requerirá de la inclusión de una interfaz que permita al usuario realizar una desambiguación del término de lo más fácilmente posible.

En esta fase se realizarán:

- Pruebas de creación de formularios que permitan guardar y cargar datos desde la base de datos de WordPress
- Creación de página de ajustes propia del plugin que muestre las etiquetas y categorías incluidas en el sitio web de WordPress
- Creación de un formulario con esas etiquetas y categorías que permita añadir manualmente un identificador de elemento de Wikidata
- Creación de un script que se conecte a la API de Wikidata y realice la llamada correspondiente a la búsqueda por el término de la etiqueta o categoría y que retorne los datos de forma estructurada para su uso
- Creación de una interfaz que muestre esos datos estructurados y permita al usuario desambiguar el término, al tener a su disposición información suficiente sobre cada uno de los elementos de Wikidata encontrados por término

#### 4.6.3 Fase 3: construcción

Una vez tenemos una aplicación capaz de proporcionar un sistema para buscar los elementos de Wikidata relacionados con el nombre de cada etiqueta o categoría, debemos decidir qué elementos guardamos y cómo los guardamos.

En esta fase se realizarán:

- Decisión del lugar de almacenamiento de los datos creados
- Decisión de la relación que tendrán los nuevos datos con las etiquetas y las categorías
- Guardado de los datos

- Prueba de carga y almacenamiento de los datos en las tablas correspondientes

#### 4.6.4 Fase 4: transición

Esta última fase comprenderá la implementación de las funcionalidades del plugin con los nuevos datos introducidos, es decir, añadirlos a los metadatos de la Web en función de los requisitos de la aplicación e incluir los enlaces a los archivos de etiquetas y categorías dentro de un microformato.

En esta fase se realizarán:

- Añadido de metadatos a la web con información sobre los elementos de Wikidata relacionados con las etiquetas y categorías en las páginas de archivo correspondientes
- Cambio del título de las páginas de archivo por un enlace al elemento de Wikidata relacionado
- Introducción de los enlaces a páginas de archivo de etiquetas y categorías en una estructura de microformatos que incluya el enlace al elemento de Wikidata relacionado

# 5 Etapa de Diseño

El presente apartado se centra en la etapa de diseño de la aplicación. Etapa en la que se recogerán los distintos aspectos que deberán ser tenidos en cuenta durante la implementación.

## 5.1 Diagrama de casos de uso

El siguiente diagrama muestra los casos de uso de un usuario típico del plugin, con el rol de administrador o propietario de la Web gestionada con WordPress.

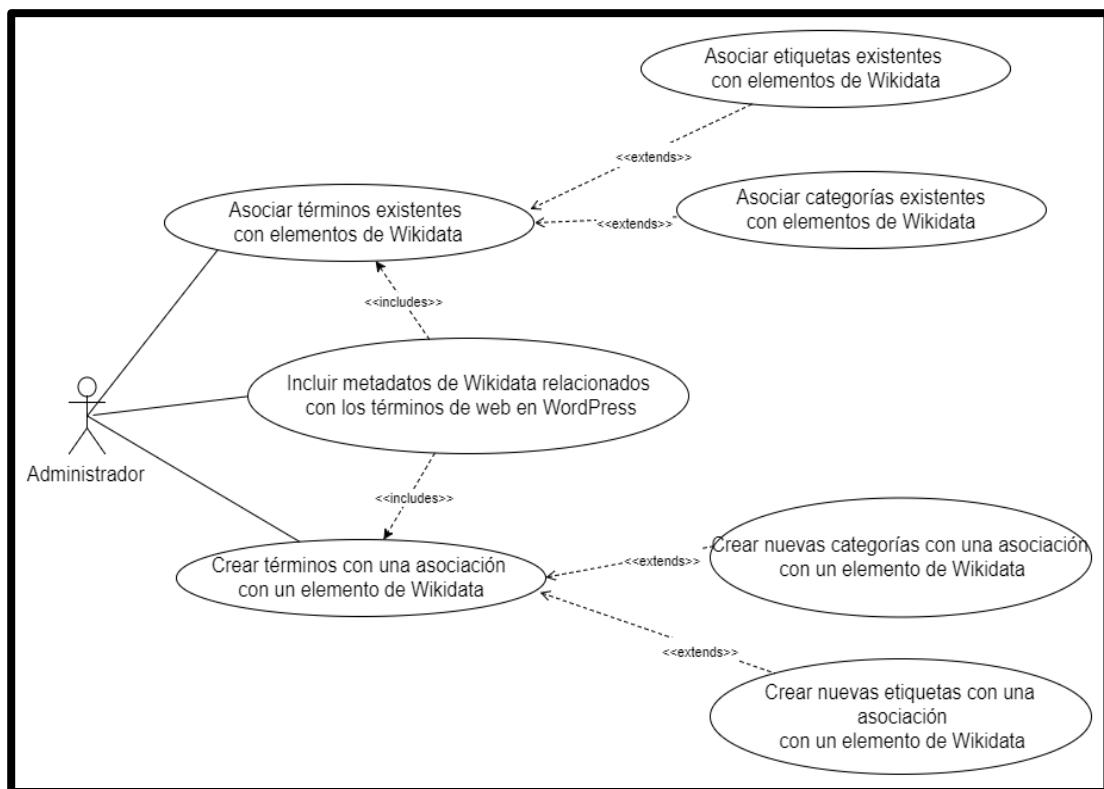


Figura 5 Diagrama de casos de uso

En la figura 5, se muestran las actividades que puede realizar el usuario del plugin, ya sea editar términos existentes añadiéndoles una asociación con un elemento de Wikidata, crearlos con esa asociación, o bien, incluir metadatos de Wikidata en la Web, lo cual necesitaría de un establecimiento de asociaciones previo.

## 5.2 Diagrama de secuencia

Dada la naturaleza del funcionamiento de WordPress, cuyo paradigma de diseño es la orientación a eventos, se presenta el siguiente diagrama de secuencia que indica en qué puntos del bucle de WordPress actúa el plugin para añadir sus funcionalidades. En el siguiente diagrama se muestra en qué puntos del bucle se anclará el plugin para añadir datos en la vista de usuario de la Web:

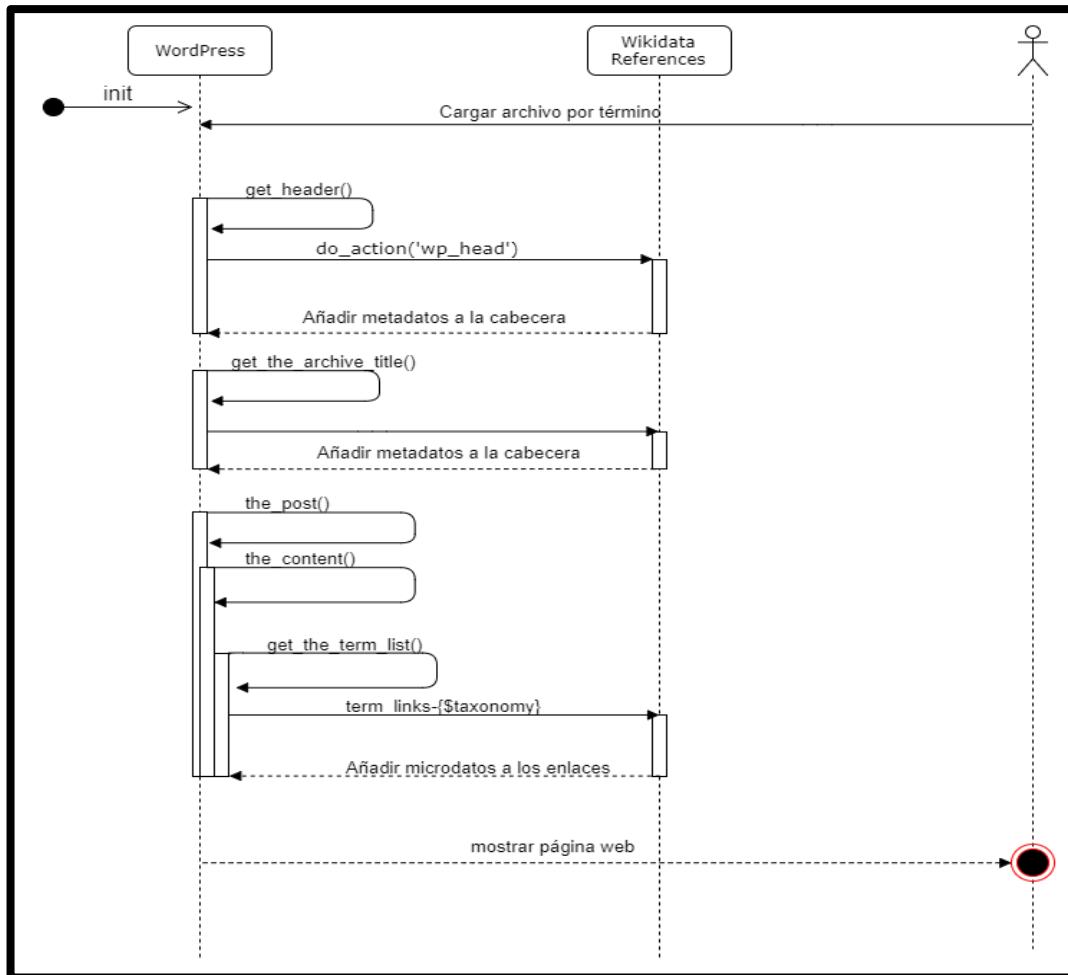


Figura 6 Diagrama de Secuencia 1: puntos de acceso del plugin en la muestra de la vista de usuario de WordPress

En el siguiente diagrama se muestran los eventos del bucle con los que se activa el plugin para añadir cambios a la interfaz de edición de etiquetas y categorías de WordPress.

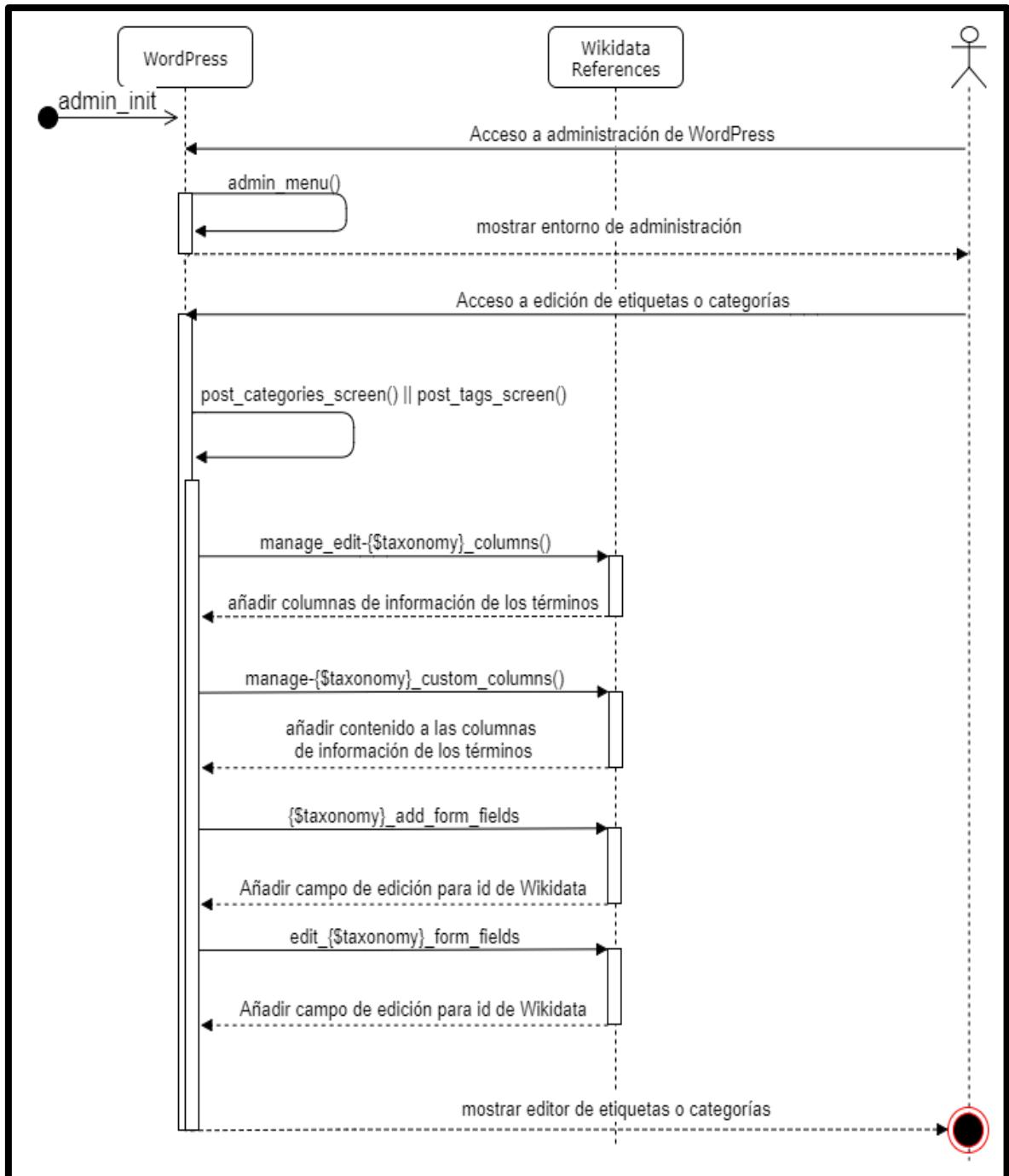


Figura 7 Diagrama de secuencia 2: puntos de acción del plugin en la página de administración de WordPress

Los eventos en los que actuará el plugin son:

- **manage\_edit-{\$taxonomy}\_columns()**: Evento generado cuando el bucle de WordPress genera la tabla de información de etiquetas y categorías y le añade columnas, por defecto, de su nombre, "slug", y descripción

- **manage\_{\$taxonomy}\_custom\_columns()**: Evento generado cuando el bucle de WordPress añade contenido a las filas de la tabla con la información de cada etiqueta o categoría
- **{\$taxonomy}\_add\_form\_fields()**: Evento generado al mostrar el formulario de creación de nuevos términos para la taxonomía indicada
- **edit\_{\$taxonomy}\_form\_fields()**: Evento generado al mostrar el formulario de edición de un término ya existente para la taxonomía indicada

En este último diagrama, se muestran puntos en los que el plugin intervendrá durante la creación o edición de un nuevo término para las diferentes taxonomías, desde el punto de la pantalla de administración de las mismas:

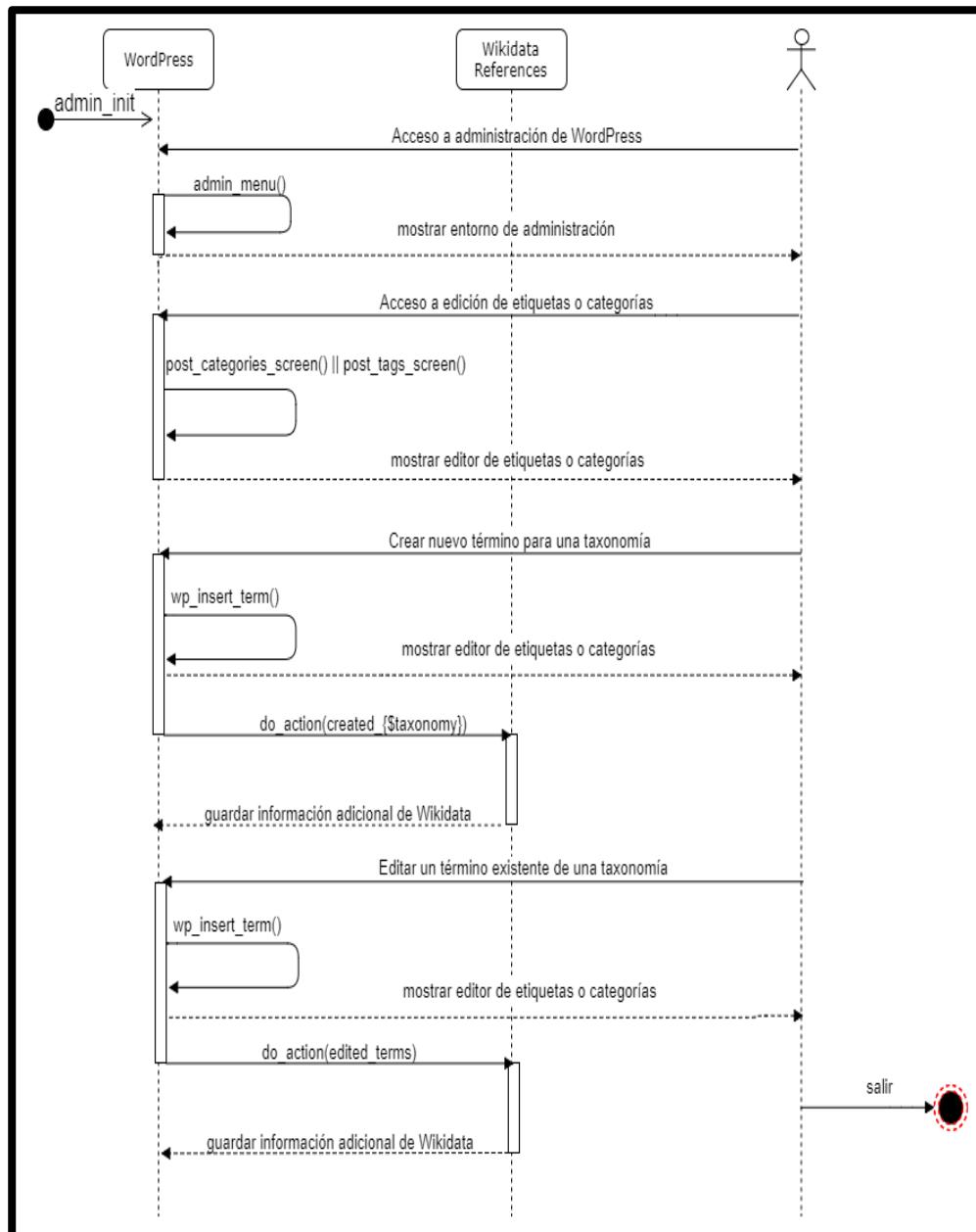


Figura 8 Diagrama de secuencia 3: creación o edición de un nuevo término para diferentes taxonomías

### 5.3 Interfaz

Toda la parte que muestra la información y generación de formularios para la edición de los términos formará parte de la propia interfaz de WordPress, integrándose de forma natural con ella. Para ello, en la página de información de términos de las taxonomías (etiquetas y categorías) se añadirá una columna con la información relativa al término de Wikidata con el que está asociado el término (tal y como exponen los diagramas de secuencia) al igual que se incluirá un campo extra en los formularios de creación y edición de términos para editar el identificador del elemento de Wikidata con el que se quiere asociar el término.

No obstante, no es posible integrar el proceso de desambiguación con la interfaz de WordPress, por lo que es necesario realizarla de otra forma.

La propuesta de diseño para este caso, supone la interacción de una ventana modal sobre WordPress que muestre las posibles opciones de desambiguación del término ajustándose al diseño de la figura 9.

### 5.4 Base de datos

Dado que estamos trabajando sobre una aplicación que va funcionar sobre WordPress y que éste tiene su propia base de datos, no tiene sentido que añadamos una propia de ninguna manera ya que además, el diseño de la base de datos del CMS ya tiene en cuenta la posibilidad de que extensiones del mismo hagan uso de la suya [6].

Este diagrama es un wireframe que muestra la estructura de la interfaz de usuario. En la parte superior, se ve la barra de navegación de WordPress con los iconos de menú y la opción 'My WordPress Page / Admin'. A la izquierda, hay un menú vertical con los siguientes ítems: Dashboard, Posts, Media, Pages, Coments, Appearance, Plugins, Users, Tools, y Settings. La sección central es una ventana modal titulada 'My WordPress Page / Admin' que contiene una tabla de datos. La tabla tiene tres columnas: 'Term name', 'Wikidata ID', y 'Description'. La descripción para cada fila es 'Lorem ipsum dolor sit amet, consectetur adipiscing elit.' y el ID de Wikidata es 'Q#####'. La tabla contiene siete filas, cada una correspondiente a un término 'name'.

Term name	Wikidata ID	Description
name	Q#####	Lorem ipsum dolor sit amet, consectetur adipiscing elit.
name	Q#####	Lorem ipsum dolor sit amet, consectetur adipiscing elit.
name	Q#####	Lorem ipsum dolor sit amet, consectetur adipiscing elit.
name	Q#####	Lorem ipsum dolor sit amet, consectetur adipiscing elit.
name	Q#####	Lorem ipsum dolor sit amet, consectetur adipiscing elit.
name	Q#####	Lorem ipsum dolor sit amet, consectetur adipiscing elit.

Figura 9 Diseño de la interfaz de desambiguación de términos de Wikidata

En concreto haremos uso de las tablas **wp\_termmeta** y **wp\_option**, a las que se accede de forma sencilla a través de la propia API de WordPress con las funciones:

```
update_option(string $option, mixed $value, bool $autoload = null)  
update_term_meta(int $term_id, string $meta_key, mixed $meta_value mixed $prev_value = '')
```

Es importante entender que estamos trabajando para desarrollar una funcionalidad externa añadida a una aplicación mayor, por tanto, el impacto de nuestro plugin sobre WordPress debe ser el menor posible.

Comprendiendo esto, hay que tener en cuenta que en caso de que el plugin sea desinstalado o desactivado de una instalación de WordPress, los datos del mismo se quedarán en la base de datos, aumentando en un gran número de filas la tabla **wp\_option**, haciendo su mantenimiento más complicado para el administrador de la base de datos, más aún, si no identificamos correctamente los nombres de las opciones que añadimos a la misma.

Por ello, es necesario que se mantengan unas buenas prácticas en el uso de la base de datos de WordPress [6], eligiendo nombres adecuados para las opciones añadidas por el plugin, por lo general añadiendo un prefijo igual para todas ellas, de forma que sea sencillo localizar en la base de datos las opciones de cada plugin.

Además, existe la posibilidad de ejecutar código en el momento de la desactivación y desinstalación del plugin, como no, eventos también, así que podemos definir una función que elimine todo rastro del plugin en la base de datos, respetando así la instalación de WordPress y por tanto al usuario.

# 6 Etapa de Implementación

Una vez se marcan los requisitos y los objetivos del proyecto, se procede a implementar el plugin, a la vez que se procura mantener un feedback continuo con el usuario que lo solicitó.

En este apartado, se detallan un resumen y los aspectos más importantes de la implementación del plugin, así como decisiones tomadas respecto a él y su correspondiente justificación.

## 6.1 Tiempo de desarrollo

En un primer momento y con unas especificaciones muy poco claras, se procede a estudiar el caso y la propuesta de aplicación creada en la plataforma Phabricator de Wikimedia, se desconocen el entorno y el lenguaje, así como las APIs de las dos entidades claves en su desarrollo: Wikidata y WordPress.

Se comienza entonces investigando sobre ambas plataformas, en un primer lugar, haciendo pruebas con la API de Wikidata, que es la que nos va a brindar la información que necesitamos, y más adelante, instalando WordPress en un servidor local, haciendo uso de XAMPP (Apache+MariaDB+PHP+Perl) y familiarizándonos con el entorno de administración del CMS. Posteriormente se realizan pruebas de desarrollo de plugins para WordPress sin mucho contenido, de forma que el uso de los ganchos y eventos en su ciclo de ejecución sean más familiares, pasando después a la elección de un framework para el desarrollo del plugin, y al propio desarrollo del mismo. Siguiendo con el método de desarrollo de software Open Up descrito en el apartado de metodología y plan de trabajo, se procede a una progresiva implementación de pequeñas funcionalidades hasta la finalización de la aplicación.

En todo caso la comunicación con el “cliente” es continua y se procura que el producto se ajuste a las necesidades especificadas y los cambios continuos o alteraciones en los requisitos, o la liviana especificidad con la que se piden algunos, llevan al proceso de desarrollo a aplicar continuos cambios de pequeños aspectos del funcionamiento del plugin.

Contando todos los procesos, estudio de tecnologías, aprendizaje de conceptos, prácticas, implementaciones y re-implementaciones, el tiempo de trabajo completo ha sido de 4 meses, con una dedicación semanal de entre diez y quince horas en los procesos de estudio y diseño en las primeras semanas, hasta entre veinticinco y treinta horas las últimas semanas de desarrollo, tal y como se puede observar en la siguiente figura.



Figura 10 Tiempo dedicado al TFG

Como se puede observar, en las últimas semanas de desarrollo del proyecto se han aumentado, de manera considerable, las horas de trabajo invertido en el mismo. Esto se debe tanto a la redacción de la presente memoria como a la corrección de pequeños detalles del código para su publicación en el repositorio de WordPress.

En total, se estiman unas 330 horas de trabajo, situándose dentro de los valores normales estipulados por el plan Bolonia, cuyo crédito ECTS (European Credit Transfer and Accumulation System) oscila entre veinticinco y treinta horas de trabajo relativo al estudiante por cada crédito.

## 6.2 Proceso de implementación

Desde el principio, unos requisitos no demasiado específicos por parte del cliente, conlleva que se deban tomar ciertas decisiones de diseño durante la implementación, aumentando así el coste de tiempo del mismo debido a los continuos cambios en el mismo, que en la mayoría de los casos se debían propagar en gran parte del código.

En un primer lugar, se elige el marco de desarrollo (en inglés framework) “*The WordPress Plugin Boilerplate*”<sup>5</sup> que proporciona una estructura orientada a objetos para el desarrollo de plugins de WordPress de calidad.

No obstante, es común enfrentarse a plugins de WordPress “profesionales” que están formados por un solo fichero de más de diez mil líneas de código, acabando así con los estándares de programación modernos.

<sup>5</sup> *The WordPress Plugin Boilerplate*: <http://wppb.io/> [último acceso: 14 de noviembre de 2017]

Tras ello, se comienzan a realizar pruebas sobre acciones y filtros, añadiendo contenido a los posts en el momento en el que son guardados, trabajando también con una página de ajustes para el plugin y con la gestión de opciones dentro del mismo.

### 6.2.1 Buenas Prácticas

El repositorio de WordPress alberga más de cincuenta y tres mil plugins, e ignorando completamente el rendimiento de nuestro sitio Web, no existe un número máximo de plugins a instalar en una instancia de WordPress, es por ello que hay que seguir una serie de buenas prácticas al desarrollar extensiones para este CMS [14].

Debido a esa posibilidad de instalar varios plugins, hay que entender el funcionamiento de los mismos dentro de WordPress. Por defecto, todas las variables, funciones y clases se definen en el espacio global de nombres (en inglés: global namespace) lo que significa que las funciones y variables que creamos pueden sobrescribir a otras ya existentes y viceversa, por tanto, hay que elegir nombres adecuados para las funciones y variables.

Inicialmente, esto se consigue no eligiendo nombres genéricos para nuestras funciones o variables. Por ejemplo, un método llamado `save_options()` tiene un nombre demasiado genérico y será sobrescrito con facilidad si se instala otro plugin que no siga estos estándares de codificación. Si además de evitar estos nombres genéricos, añadimos un prefijo a los nombres de funciones y variables, reducimos la posibilidad de que los nuestros sean sobrescritos, siendo este prefijo normalmente común a todas las funciones y variables del plugin en cuestión.

Además, PHP ofrece métodos que ayudan a disminuir este tipo de errores, en el caso de las funciones, el uso de `function_exists()` permite comprobar si ya existe una determinada función o `isset()` para variables, que evitarán que sobrescribamos o tomemos valores no esperados.

Hay que comprender también, que WordPress ya tiene su propia base de datos, con una estructura que permite el guardado de información para nuestros plugins en función de lo que necesitemos, si bien en alguna ocasión podemos necesitar crear tablas nuevas, en la mayoría de los casos, las tablas ya creadas serán suficientes [5], [9]. Es necesario también hacer un uso responsable de la base de datos, por ejemplo, almacenando las variables que guardan el estado de los ajustes de nuestro plugin en un array en lugar de añadir una fila nueva por cada opción. También, es necesario respetar la instalación de WordPress en caso de que se elimine el plugin, añadiendo una función que se ejecute al desinstalarla y que limpie los registros de la base de datos que ha creado.

Por último, dentro de las opciones de seguridad de WordPress, se deben añadir comprobaciones de permisos de usuario, funciones como `current_user_can()` o `is_admin()` nos permiten en todo momento tener control sobre quién y desde dónde se pueden ejecutar las funciones de nuestro plugin. Pero incluso esto puede no ser suficiente, además de permiso, hay que verificar la intencionalidad, tarea que nos facilita con la inserción de semillas (en inglés: nonces) que aseguran que las peticiones al servidor vienen desde la propia Web y no desde un lugar externo [12].

### 6.2.2 Recuperación y muestra de términos en los ajustes del plugin

Se llega al punto en el que es necesario avanzar, y antes de asociar las etiquetas y categorías con elementos de Wikidata, necesitamos tenerlas presentes para utilizar su nombre como término de búsqueda, y para crear la interfaz que ayude al usuario a asociarlos.

En este punto, se genera el PHP de la página de ajustes, en gran parte formado por HTML, y a través de la API de WordPress se recuperan sus etiquetas y categorías para mostrarlas en los ajustes. Es aquí donde por simplificar el diseño de la página de ajustes, se incluye la versión simplificada de carga/instalación (en inglés bootstrap).

En esta versión de la página de ajustes, se muestra un formulario con los nombres de las etiquetas, un campo vacío y un botón de búsqueda.

Term	Wikidata ID#	Search
caja	<input type="text"/>	
lampara	<input type="text"/>	
universidad	<input type="text"/>	
wikimedia	<input type="text"/>	
dubai	<input type="text"/>	
madrid	<input type="text"/>	
valencia	<input type="text"/>	
zaragoza	<input type="text"/>	
google	<input type="text"/>	
ordenador	<input type="text"/>	
wikidata	<input type="text"/>	

Figura 11 Página de ajustes

### 6.2.3 Recuperación de información de Wikipedia

Tras investigar el funcionamiento interno de WordPress y familiarizarnos con él, incluyendo curiosidades como que no es posible realizar una llamada mediante la API REST a tus propias variables, sino que debes utilizar el bucle y la API de WordPress para guardarlas, se comienza a trabajar en la funcionalidad del plugin que permite asociar términos con Wikidata, mostrando una interfaz al usuario para que pueda desambiguarlos.

Entendiendo el proceso del bucle de WordPress, y las guías de diseño y desarrollo que promulgan que sólo debería haber un botón de enviar (en inglés submit) por pantalla de ajustes, y la imposibilidad de hacer llamadas asíncronas al servidor, nos queda como opción, la programación JavaScript de las funciones deseadas.

Así, la finalidad de esta secuencia de comandos (en inglés script) sería la de tomar valores de campos específicos del formulario (el nombre de las etiquetas), la conexión a Wikidata realizando la llamada a su API, la muestra de la interfaz de desambiguación al usuario y finalmente, el rellenado del campo correspondiente del formulario con el identificador (ID) del elemento de Wikidata elegido.

¡Ya está disponible WordPress 4.9! Por favor, actualiza ahora.

Nombre	post_tag name	Wikidata ID#	Description
Google	Google	Q95	American multinational Internet and technology corporation
Google	Google	Q9366	internet search engine developed by Google
Android	Android	Q84	mobile operating system by Google
Google Earth	Google Earth	Q42274	virtual map program developed by Google
Google Maps	Google Maps	Q12013	web mapping service application by Google
google	google	Q1156923	transitive verb, meaning to search for something using the Google search engine
Google Books	Google Books	Q206033	service from Google
Knowledge Graph	Knowledge Graph	Q648625	knowledge base used by Google to enhance its search engine's search results
Google Play	Google Play	Q79576	digital application distribution platform for Android
Google+	Google+	Q356	interest based social networking service

Figura 12 Implementación de la interfaz de desambiguación de términos

dubai	Wikidata ID#	<input type="button" value="🔍"/>
madrid	Q2807	<input type="button" value="🔍"/>
valencia	Wikidata ID#	<input type="button" value="🔍"/>
zaragoza	Wikidata ID#	<input type="button" value="🔍"/>

Figura 13 Detalle del formulario de asociación de términos con identificadores de elementos de Wikidata

Una vez relleno el campo, el botón de guardado de ajustes permitiría almacenar los datos del formulario en la base de datos.

#### 6.2.4 Cambio de paradigma

Una vez aplicada la página de ajustes del plugin, y añadidos los metadatos correspondientes a las cabeceras de las páginas Web durante su carga, interviene en la tarea de Phabricator el usuario Philipjohn21<sup>6</sup> como desarrollador de WordPress y comenta que la página de ajustes está duplicando funcionalidades de WordPress, ya que, en sí, está añadiendo datos a los términos y que, por tanto, esta acción debería llevarse a cabo en la pantalla de administración de términos.

Efectivamente, tras una breve investigación se pone en claro la certeza de la intervención de Philip en el desarrollo y mi inexperiencia como desarrollador de WordPress, por lo que se comienza a rediseñar el proyecto para que la nueva información de las etiquetas, aparezca y se pueda editar en la página *edit-tags.php* de WordPress.

En estos momentos, la familiarización con la API de WordPress es mayor, y gracias a ella y a un buen diseño previo, el cambio de la interfaz a la página de edición de ajustes de etiquetas y categorías no supone más de tres días de trabajo ni cambios adicionales en la mayor parte del código.

Trabajando de este modo gracias a la acertada opinión de Philip, se consigue que la funcionalidad del plugin sea mucho más eficiente, y su integración con WordPress mayor, generando menos carga y siendo más usable para los usuarios.

### 6.3 Estado final

Finalmente se consigue un producto que se ajusta a las especificaciones y que cumple con las necesidades del usuario que pidió su desarrollo, integrándose con WordPress y sin generar errores o cambios en la instalación base del mismo.

Se consigue que el proceso de asociación de términos forme parte de la propia edición o creación de términos en etiquetas o categorías, facilitando al administrador su gestión, sin duplicar tareas en la página de administración de WordPress y sin crear confusión sobre dónde hay que asociar los términos.

La página de ajustes permite elegir exactamente qué elementos del plugin se quieren activar, incluso es posible dejar activada tan solo la asociación de términos para gestión interna.

Así, se completa el desarrollo del proyecto, y de un plugin para WordPress que a la vez que cumple con sus funciones, es respetuoso con el usuario y con el entorno en el que es instalado, no interfiriendo o cambiando variables que puedan ser utilizadas por otras funciones propias de WordPress.

---

<sup>6</sup> Wikimedia user: Philipjohn21: <https://phabricator.wikimedia.org/p/Philipjohn21/>

## 7 Conclusiones y trabajo futuro

Durante todo el proyecto se ha procurado estar en contacto con el usuario creador de la tarea lo máximo posible, si bien esto ha resultado complicado debido a la distancia geográfica que nos separa y a que éste, debido a su agenda, no contestaba a las cuestiones propuestas desde el lado del desarrollo al tiempo que éste iba evolucionando.

También se ha hecho evidente la falta de experiencia con las tecnologías con las que se ha trabajado, PHP, y las APIs de WordPress y Wikidata eran desconocidas para mí hasta el momento de inicio de este proyecto. Obviamente, los procesos mostrados en la implementación no han sido inmediatos, y en la mayoría de los casos han supuesto horas de testeo solo para comprobar el funcionamiento del bucle de WordPress.

El momento clave del desarrollo es el momento en el que aparece el apoyo de un desarrollador de WordPress, como se ha comentado anteriormente, que aconseja acertadamente sobre el diseño del plugin y sus funcionalidades, permitiendo hacer un rediseño del plugin que se integre mejor con el CMS.

Pese a que el plugin no muestra demasiados textos, se ha traducido al español utilizando la herramienta POEdit (Portable Object Template) que facilita la traducción de cadenas de texto mostradas por el código, generando un archivo .pot [2].

Para la gestión de versiones del código se ha utilizado la plataforma GitHub, y es posible acceder al código del plugin en el repositorio. Además, se ha publicado el plugin en la tienda de plugins de WordPress que ayudará a su distribución, dando la seguridad a los usuarios de que es de confianza:

 <b>GitHub</b>	<a href="https://github.com/zeko3991/Wikidata-references">https://github.com/zeko3991/Wikidata-references</a>
 <b>WORDPRESS</b>	<a href="https://es.wordpress.org/plugins/references-for-wikidata">https://es.wordpress.org/plugins/references-for-wikidata</a>

Tabla 3 URL's de acceso

Como trabajo futuro se propone: (i) el mantenimiento y la revisión de posibles incompatibilidades con otros plugins o plantillas de WordPress. En concreto, aquellas que modifiquen el contenido que cambia Wikidata References, no en el caso de las cabeceras, pero sí en el caso de los enlaces a las páginas de archivo de los términos y el título de las páginas de archivo, (ii) la actualización de la interfaz de desambiguación añadiendo la posibilidad de navegar entre páginas, ya que la API de Wikidata sólo nos permite tomar, como máximo cincuenta elementos por llamada, siempre y cuando se considerara o fuera necesario, (iii) la posible adaptación del plugin para trabajar con cualquier tipo de taxonomía, los métodos definidos en el código serían reusables, ya que funcionan independientemente de ella y, (iv) añadir un campo en el formulario de edición rápida de términos.

A lo largo del proceso de desarrollo de este TFG se ha comprendido la dificultad de trabajar con "clientes" reales, adaptando la implementación a sus cambios de opinión o peticiones, y la importancia de establecer unos requisitos mínimos y consensuados con él en la etapa inicial (análisis). En cuanto al desarrollo de aplicaciones, he aprendido a trabajar para un proyecto externo de gran magnitud, como es WordPress, ajustándome a su diseño y guía de buenas prácticas para que mi trabajo sea útil y, no afecte

negativamente ni al proyecto en sí, ni al trabajo de otro desarrollador, vislumbrando la importancia de la documentación de todo proyecto, ya que gracias a ello, la implementación del plugin ha sido independiente de cualquier interacción con la plataforma WordPress hasta el momento de su subida a su tienda de plugins.

Con respecto a WordPress, no quedan dudas de por qué es el CMS más usado actualmente, su versatilidad de cara al usuario, y el ilimitado potencial que le provee su diseño que le permite incluir todo tipo de plugins, como la API que provee y la gran cantidad de soporte que existe en la red lo convierte en un gestor de contenidos flexible y capaz de adaptarse a cualquier tipo de Web que se quiera desarrollar.

Finalmente, aunque ya se tenían conocimientos previos de SEO adquiridos durante el grado, no conocía el concepto de microdato o microformato, de gran importancia de cara a la Web Semántica.

# 8 Licencia software y documental

Copyright © 2017, Ezequiel Barbudo Revuelto

Este documento, así como sus anexos son liberados bajo los términos de una licencia GFDL.

Se permite la redistribución, copia y/o modificación de estos documentos bajo los términos de la GNU Free Documentation License, versión 1.3 o cualquier versión posterior publicada por la Free Software Foundation.

*Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation.*



El plugin liberado es software libre: puedes redistribuirlo o modificarlo bajo los términos de la GNU General Public License publicada por la Free Software Foundation en su versión 3 o cualquier versión posterior.

Este programa es distribuido con el propósito de que sea útil, pero CON NINGUNA GARANTÍA, incluso sin garantía de COMERCIALIZACIÓN o APTITÚD PARA CUALQUIER PROPÓSITO CONCRETO.

*This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.*

*This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.*



ESTE SOFTWARE ES SUMINISTRADO POR EZEQUIEL BARBUDO REVUELTO Y CUALQUIER GARANTÍA EXPRESA O IMPLÍCITA, INCLUYENDO, PERO NO LIMITANDO A: LAS GARANTÍAS IMPLÍCITAS DE COMERCIALIZACIÓN Y APTITUD PARA UN PROPÓSITO PARTICULAR, SON RECHAZADAS. EN NINGÚN CASO EZEQUIEL BARBUDO REVUELTO O COLABORADORES SERÁN RESPONSABLES POR NINGÚN DAÑO DIRECTO, INDIRECTO, INCIDENTAL, ESPECIAL, EJEMPLAR O CONSECUENCIAL (INCLUYENDO, PERO NO LIMITADO A: LA ADQUISICIÓN O SUSTITUCIÓN DE BIENES O SERVICIOS, LA PÉRDIDA DE USO, DE DATOS O DE BENEFICIOS; O INTERRUPCIÓN DE LA ACTIVIDAD EMPRESARIAL) O POR CUALQUIER TEORÍA DE RESPONSABILIDAD, YA SEA POR CONTRATO, RESPONSABILIDAD ESTRICTA O AGRAVIO (INCLUYENDO NEGLIGENCIA O CUALQUIER OTRA CAUSA) QUE SURJA DE CUALQUIER MANERA DEL USO DE ESTE SOFTWARE, INCLUSO SI SE HA ADVERTIDO DE LA POSIBILIDAD DE TALES DAÑOS.



# 9 Referencias bibliográficas

- [1]. “An ambitious Wikidata tutorial”, Wikimedia/user:Emw, 10 de Octubre de 2015, [https://commons.wikimedia.org/wiki/File:An\\_Ambitious\\_Wikidata\\_Tutorial.pdf](https://commons.wikimedia.org/wiki/File:An_Ambitious_Wikidata_Tutorial.pdf) [último acceso: 19 de noviembre de 2017]
- [2]. Cómo internacionalizar un plugin en WordPress, DECODECMS, <https://decodecms.com/internacionalizar-plugin-wordpress/> [último acceso: 17 de noviembre de 2017]
- [3]. Creating a WordPress Plugin Part 1: Adding the Admin Page, Simon Lissack, <https://blog.idrsolutions.com/2014/06/wordpress-plugin-part-1/> [último acceso: 17 de noviembre de 2017]
- [4]. Developer resources, WordPress.org, <https://developer.wordpress.org/> [último acceso: 17 de noviembre de 2017]
- [5]. How to work with WordPress Post metadata, Tom McFarlin, <https://code.tutsplus.com/es/tutorials/how-to-work-with-wordpress-post-metadata--cms-25715> [último acceso: 17 de noviembre de 2017]
- [6]. Optimizing wp-options for speed, Dave Matthews, pressjitsu.com, <https://pressjitsu.com/blog/optimizing-wp-options-for-speed/> [último acceso: 17 de noviembre de 2017]
- [7]. Orden de carga y hooks en WordPress, Fernán Díez, Betabeers.com, <https://betabeers.com/blog/orden-carga-hooks-wordpress-322/> [último acceso, 17 de noviembre de 2017]
- [8]. Porcentaje de uso WordPress entre Webs creadas con un CMS, 1&1, Los mejores CMS: una comparativa de los gestores de contenido, <https://www.1and1.es/digitalguide/hosting/cms/cms-en-comparativa-los-gestores-de-contenido-mas-usados/> [último acceso: 12 de noviembre de 2017]
- [9]. Post Meta Data Section, WordPress.org [https://codex.wordpress.org/Post\\_Meta\\_Data\\_Section](https://codex.wordpress.org/Post_Meta_Data_Section) [último acceso: 17 de noviembre de 2017]
- [10]. Usage statistics and market share of WordPress for Webstes, W3techs, <https://w3techs.com/technologies/details/cm-wordpress/all/all> [último acceso: 12 de noviembre de 2017]
- [11]. Wikidata API, Wikimedia Foundation, <https://www.wikidata.org/w/api.php> [último acceso: 17 de noviembre de 2017]
- [12]. WordPress Codex, the online manual for WordPress and a living repository for WordPress information and documentation, WordPress.org, <https://codex.wordpress.org/> [último acceso: 17 de noviembre de 2017]
- [13]. WordPress hooks: Actions, Pablo López, desarollowp.com, <https://desarollowp.com/blog/tutoriales/wordpress-hooks-actions/> [último acceso: 17 de noviembre de 2017]
- [14]. WordPress plugin development best practices, WordPress.org, <https://developer.wordpress.org/plugins/the-basics/best-practices/> [último acceso: 17 de noviembre de 2017]



# 10 Glosario de términos

**API:** del inglés: Application Programming Interface. Interfaz de programación de aplicaciones: Conjunto de subrutinas funciones y procedimientos que ofrece una librería software para ser utilizada por otros como capa de abstracción.

**Base de datos secundaria:** Una base de datos secundaria o fuente secundaria es aquella basada en el estudio y contraste de fuentes primarias fiables, siendo normalmente un comentario o análisis de las mismas.

**Bootstrap:** kit de herramientas software libre para desarrollar con HTML, CSS y JavaScript.

**Bot:** Software autónomo que es capaz de llevar a cabo tareas concretas e imitar ciertos comportamientos humanos, normalmente dedicados a procesos repetitivos.

**CC:** del inglés: Creative Commons. Organización sin ánimo de lucro, dedicada a promover el acceso y el intercambio de cultura, desarrolla una serie de instrumentos jurídicos de carácter gratuito que facilitan usar y compartir el conocimiento.

**CMS:** del inglés: Content Management System. Sistema de gestión de contenidos, software que crea una estructura de soporte para la creación y administración de contenido, principalmente en páginas Web.

**Core:** En desarrollo software, el núcleo de un software que constituye una parte fundamental del mismo.

**Framework:** del inglés: marco de trabajo. Conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problema específico y que sirve como referencia para resolver nuevos problemas similares. En desarrollo de software, es la estructura conceptual y tecnológica definida con módulos de software concretos que sirven de base para la organización y el desarrollo del mismo.

**Global namespace:** del inglés: espacio de nombres global. En programación, es un conjunto de nombres en el cual todos los nombres son únicos.

**GPL:** del inglés: “General Public License”, (Licencia Pública General). Es la licencia de derechos de autor más usada en el mundo del software libre y código abierto, garantizando a los usuarios finales la libertad de usar, estudiar, compartir y modificar el software.

**HTML:** del inglés: “HyperText Markup Language”, (lenguaje de marcas de hipertexto). Lenguaje de marcado para la elaboración de páginas Web, define una estructura básica y un código para la definición de contenido de una página Web. Es un estándar a cargo del World Wide Web Consortium (W3C).

**JavaScript:** Lenguaje de programación interpretado orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico principalmente utilizado en el lado del cliente en el desarrollo de páginas Web.

**JSON:** del inglés: “JavaScript Object Notation” (Notación de objetos de JavaScript). Formato de texto ligero para el intercambio de datos.

**Loop:** del inglés: Bucle. En programación, sentencia que ejecuta repetidas veces un trozo de código hasta que la condición asignada a dicho bucle deja de cumplirse. En WordPress, el “loop” es la secuencia del flujo de ejecución interna del mismo.

**Páginas de archivo de WordPress:** páginas donde se presentan todos los artículos publicados en una página Web WordPress que comparten una etiqueta, categoría o taxonomía.

**Phabricator:** (en la actualidad Phacility) es una suite de herramientas de aplicaciones Web para el desarrollo colaborativo de software, incluidas herramientas de revisión de código, repositorio, monitoreo de cambios y gestión de errores

**PHP:** acrónimo recursivo de “*Hypertext Preprocessor*”, (preprocesador de hipertextos). Lenguaje de programación de código abierto especialmente adecuado para el desarrollo Web y que puede ser incrustado en HTML.

**Query:** del inglés: Consulta. En informática, término utilizado para hacer referencia a una interacción con una base de datos.

**RDF:** del inglés: “*Resource Description Framework*” (Framework de Descripción de Fuentes). Familia de especificaciones del W3C originalmente diseñado como un modelo de metadatos.

**Reply Attack:** También conocido como ataque de reinyección, es una forma de ataque de red en el que en una transmisión de datos válida se inyectan datos maliciosos o se repite fraudulentamente.

**Rich snippets:** del inglés: fragmentos enriquecidos. Estructura que hace referencia a datos extraídos de páginas de resultados de búsqueda y que les da un significado semántico.

**Script:** En desarrollo software, un script es un programa generalmente simple, desarrollado para realizar tareas muy específicas.

**Semilla:** En criptografía, número arbitrario que es usado una sola vez, utilizado a menudo para protocolos de autenticación para asegurarse de que sesiones antiguas no pueden ser utilizadas en ataques de reinyección.

**SEO:** del inglés: “*Search Engine Optimization*” (Optimización para motores de búsqueda). Proceso y conjunto de técnicas mediante se cambian la estructura y la información de una página Web con el objetivo de mejorar su visibilidad en los motores de búsqueda.

**Turtle/TTL:** del inglés: “*Terse RDF Triple Language*”. Formato para serializar RDF que presenta la información mediante el uso de tripletas, las cuales constan de un sujeto, un predicado y un objeto.

**XML:** del inglés: “*eXtensible Markup Language*” (Lenguaje de Marcado eXtensible). Es un meta-lenguaje que permite definir lenguajes de marcas desarrollado por el W3C (World Wide Web Consortium).





# Anexo I: Manual de Instalación y uso de Wikidata References

“Wikidata References” es un plugin para WordPress, que permite asociar las etiquetas y categorías de tu página Web con elementos de Wikidata, para más tarde añadir metadatos y microdatos a la misma, permitiendo además anexionar enlaces a estos elementos de Wikidata en los títulos de archivo de tus etiquetas y categorías.

A continuación, se va a proceder a detallar un conjunto de instrucciones sobre el proceso de instalación y posterior uso del plugin.

## Requisitos previos a la instalación

Para poder instalar “Wikidata References” necesitas tener: (i) instalado WordPress en tu servidor, (ii) permisos de administración y, (iii) llevar a cabo los pasos que vienen detallados, a continuación, de una de las dos opciones.

## Opciones de instalación

### Opción 1:

1. Accede a <https://www.wordpress.org/plugins/references-for-wikidata/> y descarga el plugin
2. Descomprime el archivo en la carpeta /wp-content/plugins/ de la ruta de instalación de WordPress
3. Accede al panel de administración de tu WordPress
4. En la pestaña “plugins” del panel de administración, accede a “plugins instalados”
5. Busca el plugin “Wikidata References for WordPress”
6. Actívalo

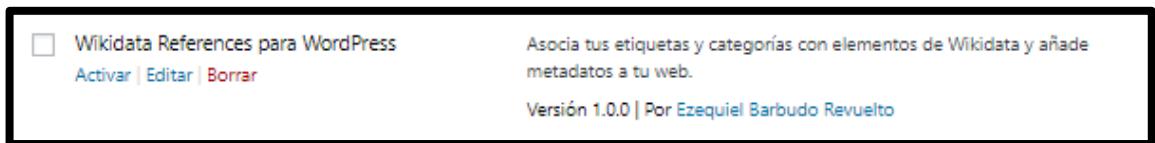


Figura 14 Opciones de instalación

7. ¡Ya está! ¡Has instalado “Wikidata References” con éxito!

### Opción 2:

1. Accede al panel de administración de tu WordPress
2. En la pestaña “plugins” del panel de administración, accede a “Añadir nuevo”
3. En el campo de búsqueda de plugins, busca “Wikidata References for WordPress”
4. Pulsa el botón “Instalar ahora”
5. ¡Ya está! ¡Has instalado “Wikidata References” con éxito!

## Manual de uso

### 1 Términos, etiquetas y categorías

El primer paso sería la asociación de las etiquetas y categorías de la Web WordPress con elementos de Wikidata. Este paso se realizará en la pestaña de edición de etiquetas y categorías, en la página *edit-tags.php* de WordPress.

Habrá que realizar el mismo procedimiento tanto para etiquetas como para categorías. Por ejemplo, en el momento de crear un nuevo término, WordPress nos presenta un formulario con varios campos, los cuales hay que llenar para crear el término.

En este caso, los que nos interesan son: "Nombre" y "Slug":

Añadir nueva etiqueta

Nombre

El nombre es cómo aparecerá en tu sitio.

Slug

El "slug" es la versión amigable de la URL del nombre. Suele estar en minúsculas y contiene solo letras, números y guiones.

Figura 15: Formulario de creación de nueva etiqueta: campos de nombre y slug

"Wikidata References" buscará en Wikidata elementos que incluyan en su título o propiedades, el "slug" de la etiqueta o categoría, mostrando sus resultados al usuario para que elija entre los encontrados cual se ajusta más al término que está asociando. De este modo, el usuario no tiene que gestionar más campos que los que administraría normalmente en los términos, y gracias a la naturaleza del "slug" de las etiquetas o categorías, el término introducido en Wikidata debería estar "limpio" de caracteres especiales que dificultaran su búsqueda.

## 2 Creación de nuevos términos para etiquetas o categorías

En el formulario de creación de términos de WordPress, “Wikidata References” añade un campo extra para introducir el identificador de un elemento de Wikidata, que aunque puede ser añadido de forma manual, el plugin permite buscarlo y añadirlo de forma semiautomática.

The image shows two side-by-side screenshots of the WordPress 'Add new tag' and 'Add new category' forms, demonstrating the 'Wikidata References' plugin. Both forms include a 'Wikidata ID' input field with a search icon and a 'Add new [tag/category]' button.

**Añadir nueva etiqueta**

- Nombre**: Text input field. Placeholder: *El nombre es cómo aparecerá en tu sitio.*
- Slug**: Text input field. Placeholder: *El "slug" es la versión amigable de la URL del nombre. Suele estar en minúsculas y contiene solo letras, números y guiones.*
- Descripción**: Text area. Placeholder: *La descripción no suele mostrarse por defecto, sin embargo hay algunos temas que puede que la muestren.*
- Wikidata ID**: Text input field with a search icon. Placeholder: *Añadir nueva etiqueta*

**Añadir nueva categoría**

- Nombre**: Text input field. Placeholder: *El nombre es cómo aparecerá en tu sitio.*
- Slug**: Text input field. Placeholder: *El "slug" es la versión amigable de la URL del nombre. Suele estar en minúsculas y contiene solo letras, números y guiones.*
- Categoría superior**: Dropdown menu set to **Ninguna**. Placeholder: *Las categorías, a diferencia de las etiquetas, pueden tener jerarquías. Podrías tener una categoría de Jazz, y por debajo las categorías Bebop y Big Band. Totalmente opcional.*
- Descripción**: Text area. Placeholder: *La descripción no suele mostrarse por defecto, sin embargo hay algunos temas que puede que la muestren.*
- Wikidata ID**: Text input field with a search icon. Placeholder: *Añadir nueva categoría*

Figura 16 Formulario de creación de términos con campo para identificador de Wikidata

Una vez completados los campos de nombre y "slug", es posible utilizar el plugin para buscar un término a través de Wikidata y, elegirlo para introducir el ID en el campo destinado a tal fin.

### 3 Edición de términos

En el caso de que la etiqueta o categoría que queramos asociar ya exista, podemos ir a su ventana de edición propia, donde habrá en el formulario, al igual que en el de creación de términos, un campo destinado para el ID de un elemento de Wikidata.

Nombre	Descripción	Slug	Cantidad
<input type="checkbox"/> z170-a		z170-a	1
<input type="checkbox"/> xampp		xampp	1
<input type="checkbox"/> workspace	<a href="#">Editar</a>   <a href="#">Edición rápida</a>   <a href="#">Borrar</a>   <a href="#">Ver</a>	workspace	1
<input type="checkbox"/> wordpress		wordpress	1
<input type="checkbox"/> Wikidata		wikidata	1

Figura 17 Detalle de la tabla de términos de WordPress en edit-tags.php

### Editar etiqueta

Nombre  El nombre es cómo aparecerá en tu sitio.

Slug  El "slug" es la versión amigable de la URL del nombre. Suele estar en minúsculas y contiene solo letras, números y guiones.

Descripción

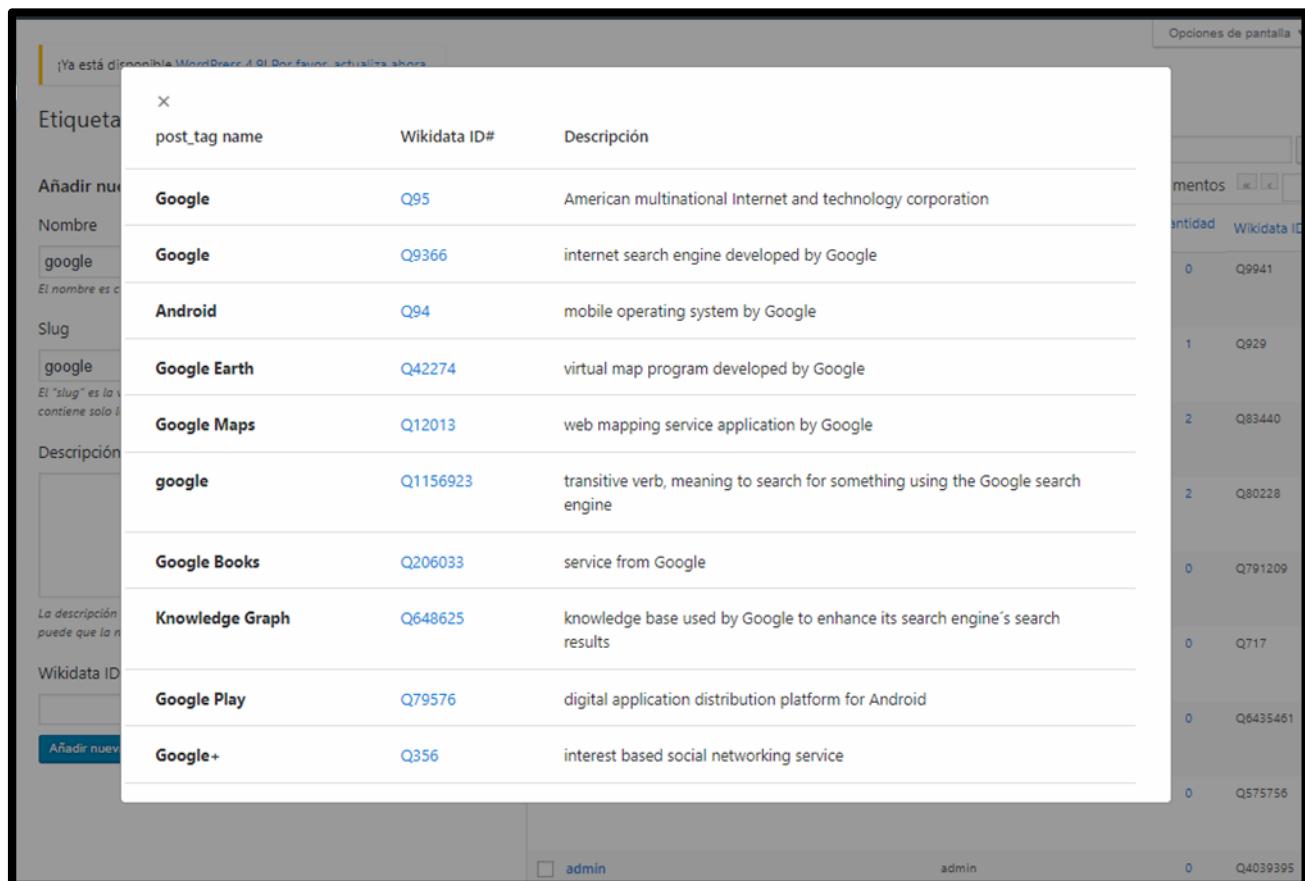
Wikidata ID

[Actualizar](#) [Borrar](#)

Figura 18 Formulario de edición de la etiqueta "wordpress" en WordPress con campo para ID de Wikidata

## 4 Asociación de términos con Wikidata

En cualquiera de los casos definidos en los puntos anteriores, el icono de la lupa situado en el lateral del campo destinado al ID de Wikidata del término, abrirá la interfaz de desambiguación del mismo, lo cual mostrará una ventana modal con un conjunto de elementos encontrados en Wikidata que contienen el término en su título o propiedades.



The screenshot shows a modal window titled 'Wikidata References' with a search bar containing 'google'. The results table has columns for 'Nombre' (Name), 'Wikidata ID#' (Wikidata ID), and 'Descripción' (Description). The results are:

Nombre	Wikidata ID#	Descripción
Google	Q95	American multinational Internet and technology corporation
Google	Q9366	internet search engine developed by Google
Android	Q94	mobile operating system by Google
Google Earth	Q42274	virtual map program developed by Google
Google Maps	Q12013	web mapping service application by Google
google	Q1156923	transitive verb, meaning to search for something using the Google search engine
Google Books	Q206033	service from Google
Knowledge Graph	Q648625	knowledge base used by Google to enhance its search engine's search results
Google Play	Q79576	digital application distribution platform for Android
Google+	Q356	interest based social networking service

Figura 19 Interfaz de desambiguación de "Wikidata References"

Al pulsar sobre el identificador de cada uno de los elementos listados en la ventana modal, se abrirá una nueva pestaña dirigida a ese elemento de Wikidata en específico. Al pulsar sobre un elemento de la lista, su identificador será añadido al campo del formulario.

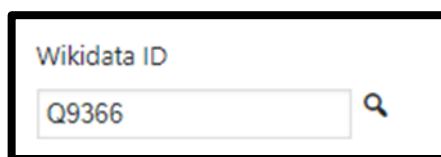


Figura 20 Detalle del campo para el identificador de Wikidata con el ID de un elemento de Wikidata

## 5 Muestra de información sobre asociaciones

En la pestaña de edición de términos, tanto para etiquetas como para categorías, “Wikidata References” añade una columna con el ID de Wikidata asociado a cada término.

<input type="checkbox"/> Nombre	Descripción	Slug	Cantidad	Wikidata ID
<input type="checkbox"/> admin		admin	0	Q4039395
<input type="checkbox"/> antiguo		antiguo	0	Q575756
<input type="checkbox"/> atencion		atencion	1	Q15757518
<input type="checkbox"/> avion		avion	0	Q791209
<input type="checkbox"/> boli		boli	0	Q717
<input type="checkbox"/> bote		bote	0	
<input type="checkbox"/> bottle		bottle	2	Q80228
<input type="checkbox"/> chrome		chrome	2	

Figura 21 Tabla de información de términos con tabla dedicada a los identificadores de Wikidata

## 6 Ajustes

“Wikidata References” permite al usuario, añadir información a su Web sobre las asociaciones realizadas entre sus términos y elementos de Wikidata de varias formas. La herramienta para gestionar esta información se encuentra en la pestaña de ajustes del plugin → WikiData References.

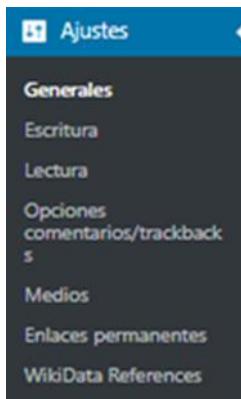


Figura 22 Pestaña de ajustes de WordPress con enlace a la página de ajustes de Wikidata References

Las opciones por las que “Wikidata References” permite añadir información a la Web WordPress son:

- Añadido de enlaces al elemento de Wikidata asociado con contexto "describedby" a las cabeceras de las páginas de archivo de las etiquetas y categorías. Siendo los formatos disponibles: HTML, json, n3, nt, php, rdf, ttl
- Cambio del título de las páginas de archivo de etiquetas y categorías, añadiéndole un enlace al elemento de Wikidata asociado para que los usuarios puedan acceder a él con facilidad
- Cambio de los enlaces a etiquetas y referencias de la Web, enmarcándolos en un microformato según los esquemas de *Schema.org*, añadiéndoles significado semántico y contextualizándolos con un enlace al elemento de Wikidata asociado

En todo caso, el usuario podrá administrar qué opciones activa y cuáles no.

¡Ya está disponible WordPress 4.9! Por favor, actualiza ahora.

## WikiData References for WordPress Setup

Wikidata references es un plugin desarrollado con el propósito de ayudarte a añadir asociaciones entre tus etiquetas y categorías de WordPress y elementos de Wikidata para añadir metadatos y microdatos en tu sitio Web. Además, te permite añadir enlaces a los términos de Wikidata asociados en las páginas de archivo de tus etiquetas y categorías.

Aquí puedes decidir que metadatos y enlaces añades a tu sitio web, para asociar tus Etiquetas y Categorías con elementos de Wikidata, accede al [editor de Términos de WordPress](#).



### Metadatos añadidos al head

Estos metadatos se añadirán al head de tus páginas de archivo de etiquetas y categorías.

Enlace al elemento de Wikidata  
<link rel="describedby" href="https://wikidata.org/entity/Q13166" type="text/html" >

Enlace al elemento de Wikidata en formato json  
<link rel="describedby" href="https://wikidata.org/entity/Q13166.json" type="application/json" >

Enlace al elemento de Wikidata en formato n3  
<link rel="describedby" href="https://wikidata.org/entity/Q13166.n3" type="text/n3" >

Enlace al elemento de Wikidata en formato nt  
<link rel="describedby" href="https://wikidata.org/entity/Q13166.nt" type="application/n-triples" >

Enlace al elemento de Wikidata en formato php  
<link rel="describedby" href="https://wikidata.org/entity/Q13166.php" type="application/vnd.php.serialized" >

Enlace al elemento de Wikidata en formato rdf  
<link rel="describedby" href="https://wikidata.org/entity/Q13166.rdf" type="application/rdf+xml" >

Enlace al elemento de Wikidata en formato ttl  
<link rel="describedby" href="https://wikidata.org/entity/Q13166.ttl" type="text/turtle" >

### Enlace a Wikidata

Esto añadirá un enlace al elemento de Wikidata asociado a tus páginas de archivo de etiquetas y categorías.

Añade un link al elemento de Wikidata en las páginas de archivo de etiquetas

Figura 23 Página de ajustes de Wikidata References

## 7 Cambios del plugin sobre la Web según los ajustes

A continuación, se mostrarán la información y los cambios que “Wikidata References” permite añadir a la Web WordPress.

### 7.1 Enlaces a Wikidata en las cabeceras HTML

Esta sección de los ajustes permite elegir qué enlaces se añaden a la cabecera de los archivos de etiquetas o categorías, pudiendo seleccionar entre diferentes formatos listados en el punto anterior.

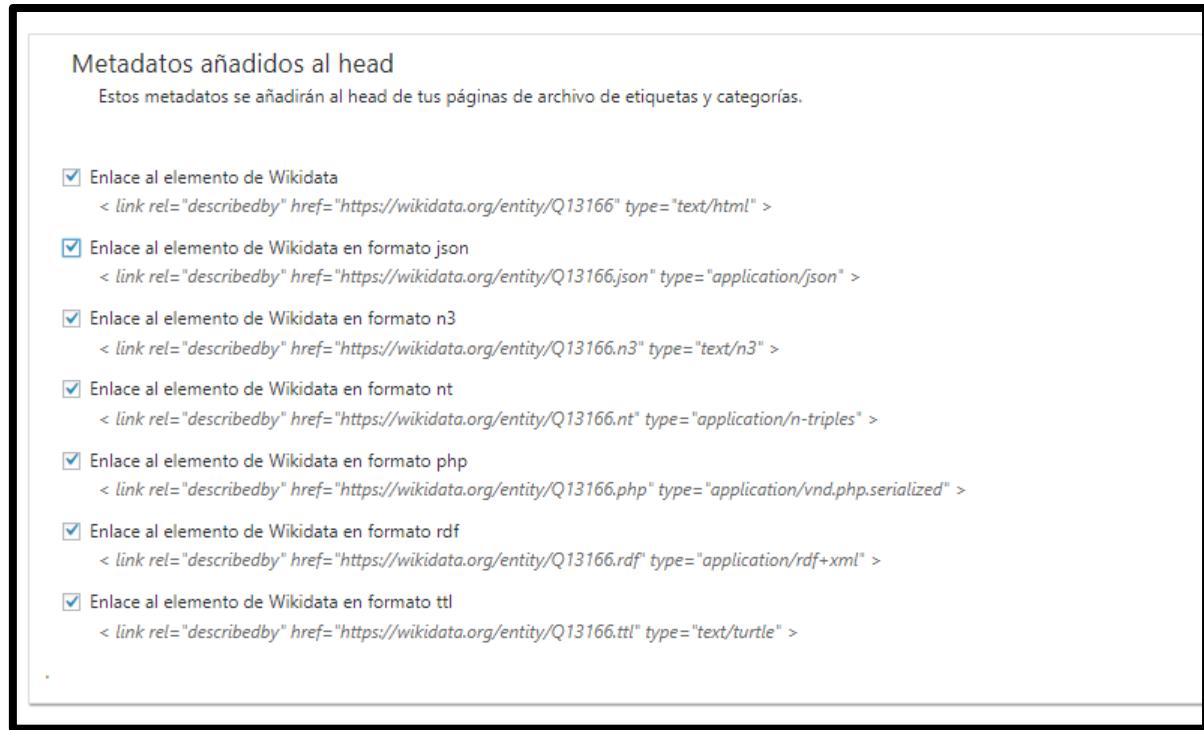


Figura 24 Ajustes de “Wikidata References” para añadido de enlaces a las cabeceras HTML de la Web

Al dejar marcadas todas las opciones de esta sección en los archivos de etiquetas y categorías, es decir, en [https://my-wordpress-site/\[tag\]/\[category\]/name](https://my-wordpress-site/[tag]/[category]/name), se añadirán los seleccionados a la cabecera:

```
<link rel="describedby" href="https://www.wikidata.org/entity/Q2013" type="text/html">
<link rel="describedby" href="https://www.wikidata.org/entity/Q2013.json" type="application/json">
<link rel="describedby" href="https://www.wikidata.org/entity/Q2013.n3" type="text/n3">
<link rel="describedby" href="https://www.wikidata.org/entity/Q2013.nt" type="application/n-triples">
<link rel="describedby" href="https://www.wikidata.org/entity/Q2013.php" type="application/vnd.php.serialized">
<link rel="describedby" href="https://www.wikidata.org/entity/Q2013.rdf" type="application/rdf+xml">
<link rel="describedby" href="https://www.wikidata.org/entity/Q2013.ttl" type="text/turtle">
```

Figura 25 Ejemplo de código HTML añadido a la cabecera de la Web por “Wikidata References”

## 7.2 Enlaces a Wikidata en el título de los archivos de etiquetas y categorías

Los archivos de etiquetas y categorías son aquellas páginas de tu WordPress que recopilan todas las entradas que has creado con una etiqueta o categoría específica, organizando así internamente la Web por contenido. Esta sección permite al usuario elegir si quiere agregar, en el título a las páginas de archivo de etiquetas y categorías, un enlace al término de Wikidata asociado.



Figura 26 Ajustes de “Wikidata References” para añadido de enlaces a Wikidata, en los títulos de archivo de etiquetas y categorías

Las opciones marcadas en la figura anterior cambiarán:



Figura 27 Página de archivo de la etiqueta “Wikidata” sin enlace

Por:

## Wiki\_workspace

Otro sitio realizado con WordPress

---

**Etiqueta: Wikidata (Q2013)**

# Entrada de WordPress

27 septiembre, 2017

avion, food, nueva  
bottle, chrome, coding,  
country, java, keyboard,  
London, madrid, mp3, open  
data, open source, pc, php,  
plugin, programming,  
Wikidata, wordpress,  
workspace

[Editar](#)

**Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id fasdfest laborum. df**

Figura 28 Página de archivo de la etiqueta “Wikidata” con enlace al elemento en Wikidata

### 7.3 Enlaces a etiquetas y categorías en microformato Schema.org

Esta opción, permitirá cambiar los enlaces a las etiquetas y categorías de tu Web, insertándolos en una estructura de microdatos de Schema.org, asociándolos con los elementos de Wikidata correspondientes.



Microformato Schema.org

Esto enmarcará los enlaces a tus categorías y enlaces en un estándar de microformato siguiendo Schema.org, añadiendo información sobre el elemento de Wikidata asociado

Enmarcar etiquetas y categorías en microformato Schema.org

```
<span itemscope itemtype="http://schema.org/Thing">
  <a itemprop="sameAs" href="http://localhost/wordpress/tag/[tag-name]" rel="tag">[tag-name]</a>
</span>
```

Figura 29 Ajuste de “Wikidata References” para cambiar enlaces a etiquetas y categorías



# ¡Hola mundo!

Bienvenido a WordPress. Esta es tu primera entrada. Edítala o bórrala, ¡y comienza a escribir!

20 octubre, 2017 / categoría [google](#), [universidad](#), [wikidata](#), [wikimedia](#), [wordpress](#) / 1 comentario / Editar

```
▼<span class="tags-links">
  <span class="screen-reader-text">Etiquetas </span>
  <a href="http://localhost/wordpress/tag/google/" rel="tag">google</a>
  ,
  ,
  <a href="http://localhost/wordpress/tag/universidad/" rel="tag">universidad</a>
  ,
  ,
  <a href="http://localhost/wordpress/tag/wikidata/" rel="tag">wikidata</a>
  ,
  ,
  <a href="http://localhost/wordpress/tag/wikimedia/" rel="tag">wikimedia</a>
  ,
  ,
  <a href="http://localhost/wordpress/tag/wordpress/" rel="tag">wordpress</a>
  ::after
</span>
```

Tabla 4 Enlaces a etiquetas sin enmarcar en microformato Schema.org

# ¡Hola mundo!

Bienvenido a WordPress. Esta es tu primera entrada. Edítala o bórrala, ¡y comienza a escribir!

20 octubre, 2017 / categoría / [google](#), [universidad](#), [wikidata](#), [wikimedia](#), [wordpress](#) / 1 comentario / [Editar](#)

```
▼<span class="tags-links">
  <span class="screen-reader-text">Etiquetas </span>
  ▼<span itemscope itemtype="http://schema.org/Thing">
    <link itemprop="url" href="https://www.wikidata.org/wiki/09366">
    <a itemprop="sameAs" href="http://localhost/wordpress/tag/google/" rel="tag">google</a>
  </span>
  ", "
  ▼<span itemscope itemtype="http://schema.org/Thing">
    <link itemprop="url" href="https://www.wikidata.org/wiki/010387767">
    <a itemprop="sameAs" href="http://localhost/wordpress/tag/universidad/" rel="tag">universidad</a>
  </span>
  ", "
  ▼<span itemscope itemtype="http://schema.org/Thing">
    <link itemprop="url" href="https://www.wikidata.org/wiki/02013">
    <a itemprop="sameAs" href="http://localhost/wordpress/tag/wikidata/" rel="tag">wikidata</a>
  </span>
  ", "
  ▼<span itemscope itemtype="http://schema.org/Thing">
    <link itemprop="url" href="https://www.wikidata.org/entity/0180">
    <a itemprop="sameAs" href="http://localhost/wordpress/tag/wikimedia/" rel="tag">wikimedia</a>
  </span>
  ", "
  ▼<span itemscope itemtype="http://schema.org/Thing">
    <link itemprop="url" href="https://www.wikidata.org/entity/013166">
    <a itemprop="sameAs" href="http://localhost/wordpress/tag/wordpress/" rel="tag">wordpress</a>
  </span>
  ::after
</span>
- - - - -
```

Tabla 5 Enlaces a etiquetas enmarcadas en microformato Schema.org

## 8 Desinstalación

Para eliminar “Wikidata References” de tu WordPress, accede a la pestaña “plugins instalados” del panel de administración de WordPress, busca el plugin “Wikidata References for WordPress” y, selecciona la opción de borrar.



Atención, desinstalar Wikidata References de tu WordPress eliminará cualquier asociación entre tus etiquetas y categorías y los elementos de Wikidata, así como los datos almacenados por Wikidata References en la base de datos de tu instalación de WordPress.



# Anexo II: La veracidad de los datos de Wikidata

Como base de datos de la que se nutren diferentes proyectos, algunos del tamaño de Wikipedia, es necesario que más allá del tratamiento de los mismos, nos cuestionemos si la información que alberga Wikidata es veraz y fiable.

Partiendo de la premisa de que los contenidos de una wiki pueden ser editados por cualquier usuario, se deben tener en cuenta los siguientes aspectos:

1. Que cualquier usuario sea capaz de modificar la información de una wiki, supone que no necesariamente todos los artículos han sido escritos por un profesional del campo, pudiendo conllevar errores de concepto o fallos de interpretación de la información a documentar. Incluso se puede ir aún más lejos, que la información esté sesgada y no se respete el *principio de neutralidad de la información*, en materias tan relevantes como la biografía de un político, ya que dependiendo de quién redacte el artículo éste puede reflejar perfiles contrapuestos
2. Comunidades opuestas pueden llegar a una *eterna rivalidad* por editar un sólo campo de un elemento de la wiki. Siguiendo con el ejemplo anterior, dos grupos políticos, a favor y en contra del mismo, pueden intentar cambiar constantemente el artículo para adaptarlo a su ideología, haciendo que la información encontrada en el mismo no sea consistente en el tiempo
3. Un usuario intenta boicotear un artículo a propósito. Estas situaciones de vandalismo [5] pueden ser tan simples como reemplazar todo un artículo con una frase jocosa o de mal gusto, pero existen soluciones más sutiles, ya que si se elimina una palabra de una frase se puede conseguir que su significado cambie radicalmente, y aunque los revisores pueden darse cuenta, muchas veces estos cambios pasan desapercibidos, llevando a información errónea
4. Fuentes mal recopiladas, ausencia de las mismas, o artículos no ceñidos a las mismas. Es posible que un artículo en una wiki no refleje adecuadamente la información de la fuente, que no exista e incluso que sea inventada. Problema este último, el más grave, ya que sin una fuente con la que contrastar la información proporcionada no es posible verificar su fiabilidad
5. Falta de fiabilidad en general de la información, debido a: (i) los casos comentados con anterioridad, (ii) sesgos no imparciales, (iii) volatilidad en la documentación, (iv) posibles vandalismos y, (iv) falta de fuentes reconocidas para contrastar la información

Para finalizar con este apartado, comentar que, en el caso de Wikipedia, una de las principales wikis que hacen uso de Wikidata, hay 5 normas oficiales conocidas como "Los cinco pilares" que establecen unas bases de comportamiento y edición de los artículos. Dichos pilares, grosso modo, vienen a exponer que Wikipedia pretende ser una enciclopedia que: (i) busca la neutralidad en la información, (ii) es de contenido libre bajo una licencia Creative Commons CC BY-SA 3,0 y (iii) al estar formada por la comunidad, es necesario el respeto entre los miembros de la misma aunque no estén de acuerdo en sus posturas, siendo el propósito llegar a un consenso neutral y real en la información aportada a los artículos. Puesto

que el propósito de una wiki es crear una fuente de conocimiento global y colaborativa, y ya que nos podemos encontrar con las desagradables situaciones comentadas con anterioridad, me gustaría hacer hincapié en que este TFG mantiene unas normas y un sistema de revisión de artículos que protege la información de cambios irrelevantes, vandalismo y/o edición y corrección de los mismos.

## 1 Principios de desarrollo de artículos

1. **Neutralidad:** Implica reflejar de manera justa, proporcionada y sin arbitrariedades toda la información significativa que haya sido publicada por fuentes fiables, incluyendo puntos de vista diferentes, llegando a un consenso y, evitando que existan artículos contradictorios y parciales sobre un mismo tema. Éste es uno de los principios fundamentales e innegociables de Wikipedia, por su naturaleza, y al ser una representación del saber humano que debe albergar todos los puntos de vista y toda la información conocida de la forma más equitativa posible, sin tomar partido por ninguna opinión sobre ningún tema en concreto. Por esta filosofía, Wikipedia busca la neutralidad y no la objetividad, ya que ésta sólo alberga un punto de vista único
2. **Verificabilidad:** Todos los artículos deben tener referencias suficientes, de forma que sea posible verificar la exactitud y neutralidad del mismo, garantizando la fiabilidad del contenido y, citando de forma adecuada la fuente con un formato estándar
3. **Fuentes fiables:** Esas referencias en cada artículo, deben ser fiables y verificables, aceptándose como tal los trabajos o documentos de autoría acreditada y profesional, y no las opiniones de los editores de la Wiki, no siendo aceptadas como fuentes otros artículos de la wiki (aunque dentro del mismo haya referencias internas). Como fuentes primarias de información, entendemos libros, revistas o archivos multimedia que son creíbles al estar publicados bajo un proceso que ofrece seguridad respecto a su veracidad, ya sea porque están respaldadas por una institución académica, divulgativa, científica o de investigación (comprenden estos campos, pero no limitados a éstos). La falta de fuentes fiables disponibles puede ser una razón para eliminar un artículo. También puede ser necesario tener en cuenta los tipos de fuentes existentes [3]:
  - a. **Fuentes primarias:** Son textos o documentos temporalmente cercanos al tema en cuestión
  - b. **Fuentes secundarias:** Son fuentes que hacen una síntesis, comentario o análisis de las fuentes primarias, generalmente agrupando varias de ellas. El uso único de fuentes secundarias no es indicador de la validez de una investigación, ya que, en muchos casos, implica que no se ha realizado una búsqueda en profundidad de las fuentes primarias
  - c. **Fuentes terciarias:** Una fuente terciaria es un documento que selecciona y compila fuentes primarias y secundarias
4. **Derechos de autor:** Se deben respetar los derechos de autor de los editores que han colaborado en el desarrollo de los artículos, protegiéndolos dentro del marco del Convenio de Berna para la protección de obras literarias y artísticas, siendo la legislación aplicable a todos los proyectos de Wikimedia de los Estados Unidos, mientras que la mayoría de los textos de la Wikipedia son liberados bajo una licencia CC BY-SA 3,0

Aunque las normas sobre verificabilidad conllevan el uso de referencias verificables que permitan contrastar la información expuesta en Wikipedia, en algunos casos, las referencias o fuentes del artículo

pueden ser, así mismo, no verificables, dando lugar a casos de información no válida o falsa proveniente de fuentes no fiables. Así, los editores no sólo tienen que hacer una búsqueda de fuentes para los artículos en los que colaboran, sino que, además, deben asegurarse de que esa documentación provenga, a su vez, de fuentes consideradas como veraces por expertos en la materia.

Con estas normas, las wikis se mantienen y autorregulan, creando una fuente documental colaborativa, creciente y libre, que en teoría trabaja para que todo su contenido sea verídico, fiable, veraz y contrastable, abierto a ampliaciones y actualizaciones continuas, siendo los propios editores los que crean y corrigen el contenido, regidos por la regla de las tres reversiones para evitar el vandalismo o las guerras de ediciones [4]. En algunos casos, los propios administradores protegen cierto contenido, ya consensuado, sobre los artículos para evitar modificaciones innecesarias y pese a esto, esa protección puede deshacerse, para editar el contenido siempre y cuando se llegue a un consenso en la discusión del mismo.

• (act · ant) ①	08:59 9 abr 2005	Orgullomoore (discusión · contribuciones) .. (32 903 bytes) <a href="#">(-1865)</a> .. (Gracias. ¡Que te vaya bien! <---así es ¿no? revierto.) (deshacer · agradecer)
• (act · ant) ①	08:57 9 abr 2005	Darwinista (discusión · contribuciones) .. (34 768 bytes) <a href="#">(+1865)</a> .. (Pareces buena persona y agradable. Lástima que seas el típico progre que no se entera de nada. Si quitárais lo de Al-Qaeda haríais un favor a la verdad. Y tu español es bueno. Me voy que mi novia me mata!) (deshacer · agradecer)
• (act · ant) ①	08:50 9 abr 2005	Orgullomoore (discusión · contribuciones) .. (32 903 bytes) <a href="#">(-1865)</a> .. (Así es, todos son bienvenidos en la Wikipedia y generalmente se perdonan equivocaciones ya que todos somos seres humanos. Además, una de las razones porque contribuyo es para mejorar mi español.) (deshacer · agradecer)
• (act · ant) ①	08:46 9 abr 2005	Darwinista (discusión · contribuciones) .. (34 768 bytes) <a href="#">(+1865)</a> .. (Por cierto, Moore: Si ni siquiera escribes bien el español, ¿qué puñetas haces aquí contribuyendo? ¿Cualquier indocumentado que no sabe ni poner tildes adecuadamente puede atreverse a escribir aquí? BUFF) (deshacer · agradecer)
• (act · ant) ①	08:35 9 abr 2005	Orgullomoore (discusión · contribuciones) .. (32 903 bytes) <a href="#">(-1865)</a> .. (¡Qué te vaya bien! revierto.) (deshacer · agradecer)
• (act · ant) ①	08:33 9 abr 2005	Darwinista (discusión · contribuciones) .. (34 768 bytes) <a href="#">(+1865)</a> .. (Wikipedia: una basura para aficionados aburridos y sectarios. Visto lo visto, esta mierda no pasará de ser una distracción de domingueros. Ale, ya podéis revertir que yo me voy a la playita con mi novia.) (deshacer · agradecer)
• (act · ant) ①	08:26 9 abr 2005	Orgullomoore (discusión · contribuciones) .. (32 903 bytes) <a href="#">(-1865)</a> .. (revierto) (deshacer · agradecer)
• (act · ant) ①	08:19 9 abr 2005	Darwinista (discusión · contribuciones) .. (34 768 bytes) <a href="#">(+1865)</a> .. (¿Me queréis decir qué sentencia judicial ha demostrado que el atentado del 11-M fue Al-Qaeda? SI AÚN NO HA HABIDO NI JUICIO!! Eso Sí que es mentir, amigo Moore. Quizás desde EE.UU no te enteras bien.;-) (deshacer · agradecer)
• (act · ant) ①	08:15 9 abr 2005	Vivero (discusión · contribuciones) .. (32 903 bytes) <a href="#">(-1)</a> .. (→Edad Contemporánea: Revierto a edición de Orgullomoore. No neutral porque analiza un solo punto de vista. ¿cómo sabes que las manifestaciones fueron causa de resultado electoral? Eso es magia) (deshacer · agradecer)
• (act · ant) ①	08:11 9 abr 2005	Orgullomoore (discusión · contribuciones) .. (32 904 bytes) <a href="#">(-1865)</a> .. (revierto. La información se encuentra en otros artículos aparte de ser obviamente partidista.) (deshacer · agradecer)
• (act · ant) ①	08:07 9 abr 2005	Darwinista (discusión · contribuciones) .. (34 769 bytes) <a href="#">(+1865)</a> .. (Decir que "no soy neutral" sin señalar qué datos son los que tergiversa es un argumento tan pobre como decir que tú tampoco eres neutral por admirar a M. Moore. Razonad si podéis en qué no soy neutral) (deshacer · agradecer)
• (act · ant) ①	07:58 9 abr 2005	Orgullomoore (discusión · contribuciones) .. (32 904 bytes) <a href="#">(-1865)</a> .. (revierto. Darwinista—ya se te ha dicho, tus adiciones no son neutrales y la neutralidad es una necesidad en cualquier artículo de wikipedia.) (deshacer · agradecer)
• (act · ant) ①	07:49 9 abr 2005	Darwinista (discusión · contribuciones) .. (34 769 bytes) <a href="#">(+1865)</a> .. (Os reto a que señáleis algún dato falso u opinable en todo lo que añado. De lo contrario, hacéis gala de un sectarismo impropio de personas amigas de la verdad objetiva.) (deshacer · agradecer)
• (act · ant) ①	07:41 9 abr 2005	Orgullomoore (discusión · contribuciones) .. (32 904 bytes) <a href="#">(-1923)</a> .. (revierto a la última edición de Comae) (deshacer · agradecer)
• (act · ant) ①	07:38 9 abr 2005	Darwinista (discusión · contribuciones) .. (34 827 bytes) <a href="#">(+1923)</a> .. (→Edad Contemporánea) (deshacer · agradecer)
• (act · ant) ①	21:28 8 abr 2005	Comae (discusión · contribuciones) .. (32 904 bytes) <a href="#">(-2633)</a> .. (Revirto. Las ediciones de Darwinista son no neutrales, y toda su información, excesivamente prolífica para este artículo, está ya en otros artículos, en los que se trata en mayor profundidad.) (deshacer · agradecer)

Figura 30 Ejemplo de "Guerra de ediciones" en Wikipedia

Por otro lado, y desde el punto de vista del usuario, éste debe protegerse en todo momento de la posibilidad de que la información que esté consultando no sea verídica, ya no solo en el ámbito de una Wiki, sino en cualquier momento, con la facilidad para publicar cualquier contenido que dé acceso a Internet. Si todo tipo de documento no basado en fuentes fiables debería ser tomado como tal, entonces, ¿cómo asegurarnos como usuarios de que un contenido es de calidad o fiable?

## 2 La calidad de la información

La ISO 9000:2000 define la calidad como el grado en el que un conjunto de características inherentes cumple con los requisitos y, la Sociedad Americana para la calidad (American Society for Quality) define la calidad como la totalidad de funciones y características de un producto que le permite satisfacer una determinada necesidad.

Dado que lo que buscamos en nuestro caso, es la calidad de la información que nos encontramos en la Web, necesitamos conocer cuáles serían sus características inherentes. Es por ello, que aunque diversos autores han tratado la materia en profundidad, nos quedaremos con la aproximación de María Pinto [1], ya aborda la materia con ciertas convenciones universalmente aceptadas de las características deseables de la información en este medio:

- **Objetividad / neutralidad:** aunque hemos hablado de que en las Wikis es preferible la neutralidad por albergar todos los puntos de vista existentes, el objetivo es parecido, la información no debe ser sesgada o parcial, ni servir para desinformar
- **Integridad:** Debe ser segura y completa, sin modificaciones o falsificaciones
- **Utilidad:** Debe ser capaz de satisfacer las necesidades de información del usuario que la recibe

Además, incide en que la calidad de la información no es una realidad palpable y medible de forma exacta y unívoca, sino que puede percibirse en varias dimensiones, que se pueden agrupar en cuatro categorías: (i) intrínseca, (ii) contextual, (iii) representacional y, (iv) la accesibilidad.

### Calidad intrínseca

Es la calidad de la propia información por sí misma, detallada con anterioridad.

### Calidad contextual

Se refiere al contexto y adecuación a las necesidades del sistema y del usuario.

- **Relevancia:** Relacionada con la precisión, se refiere a la adecuación de la información a las necesidades de los usuarios
- **Valor añadido:** Es toda característica extra que se le da a un producto o servicio con el fin de darle un mayor valor en la percepción del consumidor. En el caso de la información, facilita su uso y permite una mayor asimilación de la misma
- **Actualidad de la información:** Determina la calidad de la información en un contexto, para ser útil, la información debe estar convenientemente actualizada
- **Cantidad de información aportada:** Relacionada con la precisión y la relevancia, más información no es necesariamente mejor, ya que un exceso de la misma puede ser contraproducente
- **Utilidad:** Referida al provecho, conveniencia o interés que se saca de ella, tiene una parte subjetiva que depende de para qué quiere el usuario la información y del tipo de perfil del mismo
- **Adecuación al usuario:** Relacionada con la anterior y, dependiendo del usuario al que debe dirigirse la información, ésta tendrá que estar formada o estructurada de manera que se adapte al usuario que la recibirá

## Calidad representacional

La forma en que se representa la información, así como todos los aspectos técnicos referidos a su estructura, engloba:

- **Tipo de formato:** Medio en el que la información está representada, ya sea físico, digital, textual o multimedia
- **Claridad:** Medida en la que la información se expone de forma sencilla y accesible, bien estructurada y fácil de comprender, de modo que su falta no sea un método para la denegación tácita o la ocultación de la información
- **Concisión:** Relacionado con la cantidad, es la exactitud y precisión en la información
- **Diseño:** La información debe estar estructurada y ordenada, de forma adecuada, para facilitar su comprensión, y por tanto su utilidad
- **Flexibilidad:** Debe ser posible modificar los datos con sencillez, adaptándolos a los cambios o a nueva información para mantenerlos actualizados en todo momento
- **Homogeneidad de los datos:** Relacionada con el diseño, toda la información debería guardar cierta estructura común que facilite el manejo de los datos independientemente de la fuente o el tema que traten

## Calidad del acceso a la información

Abarca las garantías del acceso a la información en el momento en el que sea necesaria, la accesibilidad de la misma y la seguridad del usuario al acceder a ella.

- **Tiempo de espera:** Tiempo necesario para recuperar la información requerida por el usuario, en el caso de una enciclopedia en línea como Wikipedia, el tiempo de carga de la página más el tiempo que tarda el usuario en encontrar la información que busca, que depende directamente del diseño, homogeneidad y navegación. En casos de información documentada en archivos físicos, puede depender de los procedimientos legales o burocráticos que el usuario deba realizar para acceder a la información
- **Navegación:** Medida en la que la información está estructurada de modo que facilita la búsqueda efectiva y eficiente de la documentación necesaria para el usuario
- **Seguridad:** Dimensión en la que el usuario tiene garantías de que no peligra su integridad física y jurídica en el acceso a la información

### 3 La calidad de la información digital

Tal y como se ha declarado anteriormente, y con los términos expuestos, es necesario que el usuario de información en entornos digitales adopte una serie de criterios que le ayuden a filtrar entre información válida y no válida, y a identificar la veracidad, fiabilidad y calidad de la información a la que accede. Así, el usuario se protege frente a información incompleta, desactualizada, fraudulenta o falsa.

María Pinto también pone a nuestra disposición, una serie de criterios e indicadores que ayudan al usuario a juzgar la calidad de la información digital en base a ellos:

CRITERIOS	INDICADORES
<b>Autoría</b>	Adscripción del autor Información sobre el autor Medio de contacto (e-mail) Logotipo de la organización Declaración de principios y propósito del sitio web Evaluación externa
<b>Actualización</b>	Fecha de creación Fecha de actualización Información actual y actualizada Existencia de enlaces obsoletos Existencia de enlaces incorrectos
<b>Contenido</b>	Cobertura Exactitud, precisión y rigor Pertinencia Objetividad
<b>Accesibilidad</b>	Diseño compatible con diferentes versiones navegador y resoluciones de pantalla Versiones alternativas de visualización Cumplimiento normativa WAI Impresión correcta Ayuda para la navegación y la comprensión contenidos Versiones en otras lenguas
<b>Funcionalidad</b>	Estructura lógica: tabla, menú de contenidos Pertinencia y adecuación de los títulos secciones Existencia de mapa web con enlaces Sistema de búsqueda de contenidos propios
<b>Navegabilidad</b>	Menú de contenidos Botones de navegación
<b>Diseño</b>	Elegante, funcional y atractivo Combinación de colores, formas e imágenes Tipografía textual adecuada Homogeneidad de estilo y formato

Figura 31 Indicadores y criterios de la calidad de la información digital por María Pinto

En definitiva, podemos concluir que Wikidata/Wikipedia pueden ser fuentes secundarias fiables de información. Recordemos también que una fuente secundaria no es válida por sí misma, para realizar una investigación en profundidad, pero el usuario debe ser capaz de protegerse frente a información sesgada o falsa [2], no sólo la incluida en Wikidata/Wikipedia, resolviendo si la información que está usando es o no fiable, y en todo caso, contrastarla con las fuentes primarias de las que proviene para asegurarse de que es verídica, creíble, fiable y veraz.

## Bibliografía

- [1]. Pinto, M. *“Calidad y evaluación de los contenidos electrónicos”*, <http://www.mariapinto.es/e-coms/calidad-y-evaluacion-de-los-contenidos-%20electronicos/> [último acceso: 20 de noviembre de 2017]
- [2]. Ibáñez, R. *“Falsos positivos, o la importancia de comprender la información”*, <https://culturacientifica.com/2015/10/07/falsos-positivos-o-la-importancia-de-comprender-la-informacion/> [último acceso: 20 de noviembre de 2017]
- [3]. *“Wikipedia no es una fuente primaria”*, [https://es.wikipedia.org/wiki/Wikipedia:Wikipedia\\_no\\_es\\_una\\_fuente\\_primaria](https://es.wikipedia.org/wiki/Wikipedia:Wikipedia_no_es_una_fuente_primaria) [último acceso: 20 de noviembre de 2017]
- [4]. Guerras de ediciones, [https://es.wikipedia.org/wiki/Wikipedia:Guerra\\_de\\_ediciones](https://es.wikipedia.org/wiki/Wikipedia:Guerra_de_ediciones) [último acceso: 20 de noviembre de 2017]
- [5]. Tramullas, J., Garrido, P., Sánchez-Casabón, A.I., Research on Wikipedia Vandalism: a brief literature review. En: *4th Spanish Conference on Information Retrieval CERI 2016*, Granada. ACM, 2016. DOI: 10.1145/2934732.2934748