



ugr

Universidad
de Granada

Trabajo de Fin de Grado
Grado en Ingeniería Informática

Sistema de recomendación de
artículos científicos

Autor

Sergio Muñoz Gamarra

Tutor

Juan Manuel Fernández Luna





ugr

Universidad
de Granada

Trabajo de Fin de Grado
Grado en Ingeniería Informática

Sistema de recomendación de
artículos científicos

Autor

Sergio Muñoz Gamarra

Tutor

Juan Manuel Fernández Luna

Desarrollo sistema de recomendación de artículos científicos

Sergio Muñoz Gamarra

Palabras clave: Sistema de recomendación, artículos científicos.

Resumen

El desarrollo de este Trabajo de Fin de Grado (TFG) tiene como objetivo principal la creación de una aplicación web capaz de analizar los intereses de las personas que lo utilicen y recomendar a estos, artículos científicos que se ajusten a sus intereses.

La aplicación directa de este software será la recomendación de artículos científicos que serán obtenidos a través de la participación de los usuarios y nutriéndose inicialmente además de otras aplicaciones con contenido similar.

Para obtener la información necesaria para realizar unas recomendaciones lo más fielmente posibles a los intereses de los usuarios los artículos serán analizados, su contenido será parseado y serán clasificados según el tema del que traten.

A partir del procesamiento de esta información se desarrolla un sistema de recomendación mixto, es decir, inicialmente comienza con recomendaciones a partir de items o filtrado por contenido, o sea a partir de la información sacada del análisis de los artículos. Tras valorar el usuario otros artículos el sistema de recomendación pasa a ser mixto, por una parte encontramos recomendaciones por filtrado de contenido, el explicado anteriormente y por otra parte el filtrado colaborativo, en otras palabras, se analizan las valoraciones, se encuentran usuarios con valoraciones similares y se le recomienda los artículos que a estos usuarios (su vecindario) han sido de mayor interés para ellos a partir de las valoraciones que les han dado.

Abstract

Desarrollo sistema de recomendación de artículos científicos

Sergio Muñoz Gamarra

Key words: Sistema de recomendación, artículos científicos.

The main target of this “final grade work” (TFG) is the development of a web application able to analyze the interest of users and recommend them science articles that fit their profile.

This software has as a main function recommend science articles that will be provided through the participation of users. Besides, in order to avoid start the draft without articles, they were derived from other similar platforms.

With the purpose of getting necessary information to make tailored recommendation to the specific needs, the articles will be analyzed and parsed and the content will be classified according to the topic.

The information collected is used to develop a mixed recommendation system, that is to say, It initially recommends an article based on an item or a content-based filtering. When the user values other articles, the recommender system becomes mixed. In one hand, it is possible to get a content-based filtering recommendation and on the other hand the collaborative filtering, in other words, the assessments are analyzed so as to locate users who have assessed in a similar way, and finally, recommend the article which has been more useful for them.

Yo, **Sergio Muñoz Gamarra**, alumno de la titulación *Grado en Ingeniería Informática* de la **Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación de la Universidad de Granada**, con DNI 76419942, autorizo la ubicación de la siguiente copia de mi Trabajo Fin de Grado en la biblioteca del centro para que pueda ser consultada por las personas que lo deseen.

Fdo: Sergio Muñoz Gamarra

Granada a 6 de Septiembre de 2016.

D. **Juan Manuel Fernández Luna**, Profesor del área de Ciencia de la Computación e Inteligencia Artificial del Departamento de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Granada.

Informa:

Que el presente trabajo, titulado ***Desarrollo de Sistema de recomendación de artículos científicos***, ha sido realizado bajo su supervisión por **Sergio Muñoz Gamarra**, y autorizamos la defensa de dicho trabajo ante el tribunal que corresponda.

Y para que conste, expiden y firman el presente informe en Granada a 6 de septiembre de 2016.

El tutor:

Juan Manuel Fernández Luna

Agradecimientos

A Victoria y a mi familia por todo el apoyo.

A mi tutor, Juan Manuel Fernandez Luna, por toda su ayuda.

ÍNDICE

1. Introducción.....	24
1.1 Motivación	25
1.2 Objetivos del proyecto	25
1.3 Estructura del documento.....	27
2. Sistemas de recomendación	29
2.1 Introducción.....	30
2.2 Tipos.....	30
2.2.1 Sistema de recomendación basado en contenido	30
2.2.2 Sistema de recomendación colaborativo	30
2.2.3 Sistema de recomendación híbridos o mixtos.....	31
2.3 Implementación en el proyecto	31
3. Planificación	33
3.1 Fases	34
3.1.1 Fase inicial.....	34
3.1.2 Fase de análisis	34
3.1.2.1 Especificación de requisitos.....	34
3.1.2.2 Análisis de recursos	35
3.1.3 Fase de desarrollo	35
3.1.4 Fase de prueba y corrección	36
3.1.5 Redacción de la memoria.....	36
3.2 Presupuesto	36
3.2.1 Recursos	36
3.1.2.2 Recursos hardware	36
3.1.2.2 Recursos software	37
3.2.2 Costes	37
3.2.2.2 Licencias	37
3.2.2.2 Recursos hardware.....	38
3.2.2.3 Recursos humanos	38
3.2.2.4 Coste total.....	39

4. Análisis	40
4.1 Especificación de requisitos	41
4.1.1 Requisitos funcionales.....	41
4.1.2 Requisitos no funcionales o semánticos.....	45
4.1.2 Requisitos de rendimiento.....	45
4.2 Modelos de caso de uso.....	45
4.2.1 Diagramas de caso de uso.....	45
4.2.1.1 Actores	45
4.2.1.2 Diagrama.....	46
4.4 Esquemas de caso de uso	47
5. Diseño	60
5.1 Implementación y desarrollo	61
5.2 Diseño de base de datos	61
5.3 Diseño de la plataforma	62
5.3.1 Buscador.....	62
5.3.2 Iniciar sesión.....	63
5.3.3 Crear cuenta.....	64
5.3.4 Información completa del artículo	65
5.3.5 Usuarios ya valorados por el usuario	66
5.3.6 Artículos del usuario.....	67
5.3.7 Artículos mejor valorados	68
5.3.8 Usuarios mejor valorados	69
5.3.9 Recomendaciones.....	70
6. Arquitectura	71
6.1 Arquitectura del sistema.....	72
6.1.1 Base de datos	73
6.1.2 Motor de búsqueda	76
6.1.3 Sistema de recomendación	76
6.1.4 Secciones de la plataforma.....	76
6.1.5 Distribución de ficheros	77
7. Desarrollo	80

7.1 Herramientas utilizadas	81
7.1.1 Lenguajes de programación.....	81
7.1.1.1 PHP.....	81
7.1.1.2 Java.....	82
7.1.1.3 HTML	82
7.1.1.4 CSS.....	82
7.1.2 Frameworks, Bibliotecas y herramientas	83
7.1.2.1 MySQL.....	83
7.1.2.2 Apache Mahout.....	83
7.1.2.3 Apache Hadoop.....	84
7.1.2.4 Lucene y Apache Solr.....	84
7.1.2.5 Bootstrap	85
7.1.2.6 Maven.....	85
7.2 Fases de desarrollo	85
7.2.1 Base de datos.....	86
7.2.2 Parseo web de artículos	86
7.2.3 Estructurar la plataforma web	86
7.2.4 Dinamización de la plataforma	87
7.2.5 Motor de búsqueda.....	88
7.2.6 Sistema de recomendación.....	89
7.3. Pruebas.....	92
8. Resultados	93
8.1 La plataforma	94
8.1.1 Navegabilidad.....	94
8.1.2 Secciones principales.....	95
8.1.2.1 Buscador.....	95
8.1.2.1 Recomendaciones	96
8.1.3 Secciones secundarias con puntos de interés	98
8.1.3.1 Alta usuario	98
8.1.3.2 Publicar nuevo artículo	99
9. Conclusiones.....	100
9.1 Objetivos alcanzados	101

<i>9.2 Futuras vías</i>	<i>103</i>
10. Bibliografía.....	104
<i>10.1 Referencias.....</i>	<i>105</i>
<i>10.2 Enlaces de interés.....</i>	<i>106</i>
11. Anexos	108
<i>11.1 Glosario de términos.....</i>	<i>109</i>
11.1.1 Términos	109
11.1.2 Acrónimos.....	110

ÍNDICE DE FIGURAS

IMÁGENES

Imagen 1 - Diagrama de caso de uso.....	46
Imagen 2 - Diagrama de secuencia (Alta usuario)	47
Imagen 3 - Diagrama de secuencia (Modificar datos usuario).....	48
Imagen 4 - Diagrama de secuencia (Borrar usuario).....	49
Imagen 5 - Diagrama de secuencia (Alta artículo).....	50
Imagen 6 - Diagrama de secuencia (Eliminar artículo).....	51
Imagen 7 - Diagrama de secuencia (Buscador)	52
Imagen 8 - Diagrama de secuencia (Buscador por autor).....	53
Imagen 9 - Diagrama de secuencia (Consultar detalles artículo)	54
Imagen 10 - Diagrama de secuencia (Valorar artículo).....	55
Imagen 11 - Diagrama de secuencia (Artículos mejor valorados).....	56
Imagen 12 - Diagrama de secuencia (consultar artículos valorados por el usuario)	57
Imagen 13 - Diagrama de secuencia (usuarios mejor valorados)	58
Imagen 14 - Diagrama de secuencia (Recomendaciones).....	59
Imagen 15 - Esquema Entidad-Relación.....	61
Imagen 16 - Diseño pantalla buscador	62
Imagen 17 - Diseño pantalla de inicio de sesión.....	63
Imagen 18 - Diseño pantalla de registro.....	64
Imagen 19 - Diseño de la pantalla de información sobre artículos	65
Imagen 20 - Diseño de la pantalla de artículos ya valorados.....	66
Imagen 21 - Diseño de la pantalla de artículos ya publicados por el usuario	67
Imagen 21 - Diseño de la pantalla de mejores artículos.....	68
Imagen 22 - Diseño de la pantalla de usuarios mejor valorados	69
Imagen 23 - Diseño de la pantalla de artículos ya valorados.....	70
Imagen 24 - Esquema relación entre módulos.....	72
Imagen 31 - Menú	94
Imagen 32 - Buscador	95
Imagen 33 - Recomendaciones basadas en contenido.....	96
Imagen 34 - Recomendaciones por filtrado colaborativo	97
Imagen 35 - Registro de usuarios	98
Imagen 36 - Publicar nuevos artículos	99

1. Introducción

1.1 Motivación

Actualmente podemos encontrar buscadores de todo tipo de archivos, donde podemos encontrar desde texto hasta imágenes. Existen varias aplicaciones web en las que se muestra información sobre investigaciones científicas (Google Scholar, sciencedaily, sciencenewsforstudents,...), pero ninguna que centralice la mayor parte de los artículos científicos. Por otra parte en la búsqueda de artículos científicos es especialmente importante la facilidad para encontrar información sobre el tema sobre el que tratamos, ya que hablamos de temas de innovación donde la información puede ser escasa al ser un tema de investigación novedoso sobre el que poca gente habrá escrito.

Además, el proyecto puede servir como herramienta de difusión, pues en la era de información la manera más rápida de tener difusión global es a través de internet y las revistas de divulgación no consiguen llegar a un público tan extenso como internet [5]. De este modo si existiese una plataforma que consiguiera abarcar un porcentaje importante de la comunidad científica, podría hacer que cada uno de los artículos que apareciese en la misma tomase una relevancia importante nada más llegar a la misma.

1.2 Objetivos del proyecto

El objetivo de este proyecto es la creación de una plataforma web donde los usuarios puedan publicar sus artículos de investigación, para hacerlos llegar a más gente, y además facilitar que todos los usuarios puedan llegar a tener la oportunidad de trabajar con todo el material de investigación más reciente de manera sencilla, sin necesidad de que el usuario tenga que buscarlo, pues el sistema de recomendación podrá proveérselo.

Para cubrir las necesidades que han llevado a la creación de este proyecto y poder afirmar que el proyecto cubre las mismas, se procede a la exposición de los objetivos que debe cumplir. Posteriormente, al final de la memoria se realizará un análisis de los mismos, comprobando si se ha conseguido cubrirlos todos y, en caso de no hacerlo, encontrar el por qué.

Objetivo 1	Dar de alta usuarios
Tipo	Obligatorio
Descripción	La plataforma debe permitir darse de alta a usuarios.

Objetivo 2	Introducir de artículos en la plataforma
Tipo	Obligatorio
Descripción	La plataforma ha de ser capaz de subir y registrar en la base de datos la información referente al artículo.

Objetivo 3	Buscar sobre temas
Tipo	Obligatorio
Descripción	La plataforma ha de ser capaz de encontrar artículos referentes a una búsqueda.

Objetivo 4	Permitir votaciones a artículos
Tipo	Obligatorio
Descripción	La plataforma debe permitir a los usuarios puntuar artículos con el fin de conocer sus gustos.

Objetivo 5	Realizar recomendaciones
Tipo	Obligatorio
Descripción	La plataforma ha de ser capaz de realizar recomendaciones a los usuarios a partir de sus intereses y de los gustos de otros usuarios con lo que tiene puntuaciones en común.

Objetivo 6	Listar los artículos con mejores valoraciones
Tipo	Opcional
Descripción	La plataforma deber ser capaz de listar los artículos que mejores valoraciones han obtenido por parte de los usuarios

Objetivo 7	Nutrir a la plataforma de otras fuentes al comienzo
Tipo	Opcional
Descripción	La plataforma deber ser capaz de nutrirse a partir de otras fuentes que no sean los usuarios en un comienzo para evitar que al principio la plataforma no tenga escasez de información.

Objetivo 8	Ser eficiente
Tipo	Opcional
Descripción	La plataforma debe de ser eficiente, es decir, teniendo en cuenta la gran cantidad de datos y documentos con los que puede llegar a trabajar es importante llegar a soluciones que hagan que los tiempos de carga no sean molestos para el usuario.

1.3 Estructura del documento

Este documento está compuesto de los siguientes apartados:

- **Resumen:** Breve resumen de la funcionalidad del proyecto, así como un resumen más extendido en inglés.
- **Introducción:** Pequeña introducción al proyecto, tratando los siguientes temas:
 - Motivación que ha llevado a crear el proyecto
 - Estructura básica del documento
 - Lista de objetivos a lograr durante la realización del proyecto
- **Sistemas de recomendación:** Al ser la parte más importante del proyecto se ha añadido una pequeña sección donde se explican qué son, cuántos tipos hay, y cómo encajan en este proyecto.
- **Planificación:** Exposición de la planificación del proyecto, las distintas etapas por las que pasará el proyecto y el presupuesto por el que se podría crear.
- **Análisis:** Fase en la que especifica los requisitos, casos de uso y flujos del sistema.

- **Diseño:** Fase en la que se deciden las tecnologías necesarias para el mismo, así como la interfaz y el diseño de la plataforma.
- **Implementación:** Aspectos más relevantes sobre el desarrollo del proyecto y exposición de los principales problemas que surgieron durante la implementación del mismo. Se divide en:
 - Exposición de todas herramientas y tecnologías utilizadas y el motivo por el que se eligieron.
 - Fases de desarrollo, donde se encuentran todos los pasos seguidos durante la implementación
 - Diseño de la interfaz, donde se muestran todas las pantallas de la plataforma, su utilidad y su uso.
- **Resultado:** Encontraremos reflejados los puntos principales de los resultados, con capturas de pantalla y pequeñas explicados sobre algunos puntos de interés.
- **Conclusiones:** Resumen final del proyecto explicando las conclusiones a las que se han llegado durante el desarrollo y exposición de mejoras futuras.
- **Bibliografía:** Lista de fuentes de información utilizadas durante la realización del proyecto.
- **Anexo:** Otros documentos relevantes, así como un glosario de términos y acrónimos.

2. Sistemas de recomendación

2.1 Introducción

Los sistemas de recomendación[4] son herramientas software que permite a un sistema sugerir contenido o productos de características similares, basándose en información que el sistema recoge de diversas maneras: mediante votaciones, etiquetando al usuario,.... Dichos sistemas nacieron con la finalidad de solucionar el problema de la sobrecarga de información a la cuál estamos expuestos[7].

2.2 Tipos

Los sistemas de recomendación se dividen según según el modo en el que recogen la información de la cuál partirán para realizar las recomendaciones. Se pueden encontrar divididos principalmente en tres categorías[1], que serán explicadas a continuación.

2.2.1 Sistema de recomendación basado en contenido

Estos sistemas basan su filtrado en el análisis de los intereses del usuario según los ítems que ha valorado previamente (p. ej. Youtube) o cualquier otro método por el que el usuario ha dicho de forma implícita o explícita que esta interesado en ese tema (p. ej. Google) [2].

El funcionamiento de este tipo de sistemas de recomendación conlleva una condición básica y es que cada ítem que llegue al sistema ha de ser indexada y analizada, para que de este modo el sistema de recomendación conozca el contenido y pueda catalogarlo.

2.2.2 Sistema de recomendación colaborativo

Los sistemas de recomendación colaborativos basan sus recomendaciones en el análisis de las votaciones que han realizado los usuarios, y recomienda aquellos ítems que no han sido votados por el usuario activo y que han resultado bien valorados por los usuarios similares.

La forma en la que estos SR se podría dividir en dos procesos principales:

1. El sistema de recomendación analiza todas las votaciones y crea vecindarios, es decir, grupos de usuarios que comparten un alto porcentaje de votaciones.
2. Para cada usuario, busca dentro de su vecindario elementos que no hayan sido votados por él pero sí por usuarios que están dentro de su vecindario con una puntuación alta.

2.2.3 Sistema de recomendación híbridos o mixtos

Los sistemas de recomendación híbridos son aquellos que combinan el funciona de dos o más tipos de sistema de recomendación, es decir, combinan varios tipos de algoritmo de recomendación. Estos sistemas nacieron con la finalidad de acabar con ciertas carencias que los sistemas de recomendación tienen, por ejemplo el arranque frío en el filtrado colaborativo, necesidad de conocimiento en los sistemas de recomendación con filtrado por contenido, etc.

2.3 Implementación en el proyecto

En este proyecto los sistemas de recomendación son un módulo básico. En el planteamiento del proyecto se tuvieron en cuenta todos ellos y tras el análisis se llegó a la conclusión de que la mejor solución era la utilizando de un sistema de recomendación híbrido, para evitar problemas como el inicio frío.

Para el sistema de recomendación basado en contenido se necesitaba una herramienta para realizar el análisis gramatical de los archivos que llegarían, así que se opta por la incorporación de Solr, herramienta que nos provee un sistema de perseguido completo, en el que podemos indexar toda la información para la búsqueda. Por otra parte necesitábamos una primera información que nos dijese los intereses del usuario y con la cuál pudiésemos empezar las recomendaciones basadas en contenido, pero tampoco queríamos que fuese algo restrictivo y que no se pudiese cambiar, así que optamos por que el usuario al registrarse en la plataforma introdujese palabras que describiesen el campo en el que trabaja (p. ej. Recuperación de información), el tema de su última investigación (p. ej. Sistemas de recomendación mixtos) y su tema preferido (p. ej. Motores de búsqueda). De este modo se realiza una búsqueda con estas palabras sobre el motor de búsqueda obteniendo recomendaciones basadas en el contenido.

Por otra parte encontramos el sistema de recomendación colaborativo, que basa en la herramienta de *machine learning*, Mahout, la cual tiene implementados algoritmos de recomendación y nos permite utilizar métodos para la obtención del vecindario del usuario y otras funciones para la obtención de recomendaciones. De este modo la

plataforma provee al sistema de recomendación de un fichero con las valoraciones y este los procesa generando las recomendaciones.

De este modo obtendríamos un sistema de reconvención híbrido, en el que evitaríamos el problema del inicio frío y por otra parte también daríamos dinamismo a las recomendaciones con la posibilidad de cambiar las etiquetas que se le asignan al usuario y el sistema de recomendación por contenido.

Encontrará más detalles de implementación en la sección 5.2.5 y 5.2.6.

3. Planificación

3.1 Fases

La planificación del proyecto, como en la mayor parte de los proyectos de este tipo, se dividió en varias fases, las cuáles son explicadas a continuación:

3.1.1 Fase inicial

El punto de partida es la necesidad de crear una plataforma donde se pudiesen centralizar todas las investigaciones y artículos científicos dando un valor mayor al introducir en ella un sistema de recomendación haciendo así más fácil encontrar información de nuestro interés.

Para llegar a construir una plataforma con las características anteriormente citadas se esboza una solución con la ayuda de Juan Manuel Fernández Luna y se extraen los requisitos funcionales que debería cumplir una forma de estas características. Además se decide que sea un desarrollo en cascada[6], es decir, que hasta que no se acabe una fase no se podrá empezar con la siguiente.

Tras cerrar los requisitos se llegó a la conclusión de que se necesitaría utilizar múltiples tecnologías. Por una parte tendríamos que tratar la creación y estructuración de la plataforma utilizando para ello HTML y CSS; por otra parte se necesitaba hacer dinámica la plataforma, por lo que se introdujo PHP y MySQL; además se necesita analizar los artículos que llegasen a la plataforma, por lo que se necesitaría Lucene y Hadoop; finalmente se necesitaba una herramienta para la construcción del sistema de información y para ello se llegó a la conclusión de la herramienta que mejor se ajustaba a las necesidades era Mahout. La elección de estas tecnologías se tratará más a fondo en siguientes apartados.

3.1.2 Fase de análisis

En esta etapa se estudió aquellos requisitos básicos para el correcto funcionamiento de la plataforma, así como los recursos que se necesitarían.

3.1.2.1 Especificación de requisitos

Se realizó un estudio de los requisitos que necesitaba tener la plataforma para cubrir todos los objetivos y necesidades.

3.1.2.2 Análisis de recursos

Se realizó un estudio de las tecnologías que se iban a necesitar y de las que ya hemos hablado y el estudio de la integración de las mismas.

3.1.3 Fase de desarrollo

Una de las fases más importante, extensa e intensa de todas, en la que se ha llevado el desarrollo de la plataforma.

Para facilitar las tareas y para seguir metodologías de trabajo actuales, en las que prima la eficiencia, se dividió en objetivos:

- Montar el servidor para utilizar PHP y MySQL
- Creación de la Base de Datos MySQL. Estructuración de la misma.
- Crear las páginas con HTML y CSS.
- Desarrollo de funciones PHP, para dinamizar la plataforma, ajustándonos a los requisitos.
- Instalación y puesta a punto del sistema de indexación de ficheros SOLR (Lucene y Hadoop).
- Instalación y configuración de la herramienta de machine learning, Mahout.
- Implementación de sistema de recomendación con Mahout.

Para conseguir realizar cada una de estas tareas fue necesario el estudio y adquisición de conocimientos en las herramientas utilizadas. Para concretar más, fue necesario el estudio de las siguientes herramientas:

- **Buscador**

Uno de los requisitos básicos de la plataforma es la indexación de todos los ficheros que llegan a la misma, por lo que se necesitaba una herramienta que funcionará como motor de búsqueda y que permitiese de forma sencilla conexiones para extracción de información, por lo que Solr al tener una interfaz de usuario bastante intuitiva y con facilidad para configurar el motor de búsqueda y además tener una API en Json facilitaba en gran medida esa tarea.

De esta forma también hubo la necesidad de adquirir conocimientos en Json, la obtención de información dicha API y el parseo del Json de respuesta.

- **Sistema de recomendación**

El punto principal del proyecto es el sistema de recomendación y por lo tanto se necesitaba una herramienta de machine learning que proveyese de herramientas para el desarrollo del sistema de recomendación basado en vecindario, por lo que se eligió Mahout, el cuál ya incorpora funciones de sistema de recomendación y calculo de vecinadarios.

3.1.4 Fase de prueba y corrección

En conjunto con la fase de desarrollo, tras la finalización de cada objetivo se realizaron pruebas y correcciones de errores que iban apareciendo.

3.1.5 Redacción de la memoria

Tras finalizar el todas las fases anteriores y tras haber ido guardando la información de los pasos anteriores se procedió a escribir la memoria del proyecto.

3.2 *Presupuesto*

En este apartado vamos a listar todos los recursos utilizados para la realización del proyecto, tanto hardware como software.

3.2.1 Recursos

3.1.2.2 Recursos hardware

- Ordenador usado para realizar el proyecto: MacBook Pro - 2,5 GHz Intel Core i5 - 4 GB 1600 MHz DDR3.

3.1.2.2 Recursos software

- **Sistema operativo:** OS X El capitan versión 10.11.5
- **IDE:** PhpStorm, Netbeans
- **Lenguajes de programación:**
 - PHP 5.6.10
 - Java 1.8.0_91
 - HTML5
 - CSS3
 - JavaScript
- **Diseño de diagramas:** Dia
- **Procesados de texto:** Pages

3.2.2 Costes

3.2.2.2 Licencias

Uno de los recursos que menos se tiene en cuenta al presupuestar un proyecto son las licencias, pero a su vez es importante para no disparar el precio del proyecto con estos gastos imprevistos. Para evitarlo vamos a listar las licencias que necesitaríamos con el precio y la licencia a la cuál se acoge:

- Java 1.8.0
 - Licencia: Oracle Binary Code License (BCL)
- Apache
 - Licencia: Apache License (Software Libre permisiva)
- LibreOffice
 - Licencia: LGPLv3

- OS X El capitan:
 - Licencia: APSL

- PhpStorm
 - Licencia de estudiante

Teniendo en cuenta estas licencias, el coste por software sería de: 19,90€.

3.2.2.2 Recursos hardware

Como ya ha sido mencionado anteriormente, para la realización de este proyecto se utilizó un MacBook Pro, adquirido a mediados de 2012. El equipo costó 1.049 €.

3.2.2.3 Recursos humanos

Para la realización de este proyecto se necesitarían varios perfiles profesionales:

- Desarrollador web Senior
 - Coste: Entre 25.000€ y 28.000€ / Año
- Programador Java
 - Coste: Entre 25.000€ y 28.000€ / Año
- Diseñador web
 - Coste: Entre 22.000€ y 25.000€ / Año

Estimando que el proyecto se llevaría a cabo en 6 meses, el gasto en salarios sería de: 38.250€ (tomando como salario el centro del rango salarial dado anteriormente).

3.2.2.4 Coste total

El coste total del proyecto, teniendo en cuenta tanto el hardware que se usaría para cada empleado (vamos a suponer que todos tendrían el mismo equipo) y el coste de sus salario, será el siguiente:

Descripción	Explicación	Coste total
Licencias	1 Licencia Pages	19,90 €
Hardware	1.049 € x 3 Equipos	3.147 €
Desarrollador web Senior	26.500€ / Año = 2.208 € / mes	13.250 €
Programador Java	26.500€ / Año = 2.208 € / mes	13.250 €
Diseñador web	23.500€ / Año = 1.958 € / mes	11.750 €
Coste Total		41.416,90 €

Tabla 1 - Estimación de costes

4. Análisis

4.1 Especificación de requisitos

En esta sección podremos el detalle de todos los requisitos y condiciones que proyecto debe cumplir para conseguir cubrir todas las necesidades por las que dicho proyecto ha nacido.

Todos los requisitos se identifican de forma inequívoca mediante un código que constará de una codificación indicando el tipo de requisito y un número de orden. Este código será utilizado como referencia cada vez que sea necesario mencionarlo a lo largo del desarrollo del proyecto.

4.1.1 Requisitos funcionales

Para que que la plataforma sea funcional y los usuarios puedan obtener los mejores resultados con ella debe cumplir una serie de requisitos funcionales, los cuáles se dictan a continuación:

RF1	Dar de alta usuario
Explicación	La plataforma dará de alta a un usuario, almacenando sus datos.
Datos de entrada	Datos relativos al usuario.
Datos de salida	Ninguno.

RF2	Enlazar usuarios a etiquetas
Explicación	El usuario señalará como máximo tres etiquetas sobre sus intereses.
Datos de entrada	Etiquetas y identificador.
Datos de salida	Ninguno.

RF3	Borrar usuario
Explicación	El usuario podrá borrar su cuenta, y con ello todos los artículos y votaciones realizados por su parte.
Datos de entrada	Ninguno.
Datos de salida	Ninguno.

RF4	Modificar datos de usuario
Explicación	El usuario podrá modificar los datos con los que se dio de alta.
Datos de entrada	Datos a modificar.
Datos de salida	Ninguno.

RF5	Publicar artículo
Explicación	La plataforma almacenará en base de datos todos los datos relativos al artículo, el fichero pdf del mismo y lo indexada en el motor de búsqueda.
Datos de entrada	Datos relativos al artículo y ficheros, así como el identificador del usuario.
Datos de salida	Ninguno.

RF6	Enlazar etiquetas con artículos
Explicación	El usuario podrá enlazar etiquetas como palabras descriptivas a sus artículos.
Datos de entrada	Conjunto de etiquetas y identificador del artículo.
Datos de salida	Ninguno.

RF7	Visualizar todos sus artículos
Explicación	El usuario podrá listar todos los artículos que ha subido.
Datos de entrada	Ninguno.
Datos de salida	La información de todos los artículos que ha subido.

RF8	Borrar artículo
Explicación	El usuario podrá eliminar artículos que él haya publicado.
Datos de entrada	Identificador del artículo.
Datos de salida	Ninguno.

RF9	Buscador
Explicación	El usuario podrá realizar búsquedas a partir de palabras clave.
Datos de entrada	Conjunto de palabras a buscar
Datos de salida	Información de todos los artículos relacionados con esas palabras.

RF10	Buscador por autor
Explicación	El usuario podrá realizar buscar todos los artículos subidos por un autor.
Datos de entrada	Nombre del autor.
Datos de salida	Información de todos los artículos relacionados con ese autor.

RF11	Consultar información detallada del artículo
Explicación	El usuario podrá, a partir de la búsqueda, consultar toda la información detallada del artículos, refiriéndonos con esto a etiquetas asociadas, valoraciones, etc, así como el pdf con el artículo.
Datos de entrada	Identificador del artículo.
Datos de salida	Toda la información del artículo

RF12	Valorar artículos
Explicación	El usuario podrá realizar votaciones de los artículos.
Datos de entrada	Valoración numérica, identificador del artículo e identificador del usuario.
Datos de salida	Ninguno.

RF13	Modificar valoración
Explicación	El usuario podrá modificar la votación que le dio a algún artículo.
Datos de entrada	Valoración numérica, identificador del artículo e identificador del usuario.
Datos de salida	Ninguno.

RF14	Consultar artículos mejor valorados
Explicación	El usuario podrá listar los artículos que han recibido mejores valoraciones.
Datos de entrada	Ninguno.
Datos de salida	Información de la lista de mejores artículos.

RF15	Consultar artículos ya valorados
Explicación	El usuario podrá consultar todos los artículos que ya ha valorado.
Datos de entrada	Identificador del usuario.
Datos de salida	Información de los artículos.

RF16	Consultar usuarios registrados mejor valorados
Explicación	El usuario podrá listar los usuarios que ha subido los artículos con mejores valoraciones.
Datos de entrada	Ninguno.
Datos de salida	Información de los usuarios mejor valorados.

RF17	Indexar artículos nuevos
Explicación	Al ser publicado un nuevo artículo es necesario que la plataforma lo indexe en el motor de búsqueda para que pueda ser buscado.
Datos de entrada	Fichero del artículo e identificador del mismo.
Datos de salida	Indexación del artículo.

RF18	Recomendar
Explicación	La plataforma debe ser capaz de recomendar artículos a partir de los gustos del usuario y de sus votaciones.
Datos de entrada	Identificador del usuario.
Datos de salida	Artículos que se van a recomendar al usuario.

4.1.2 Requisitos no funcionales o semánticos

RS1 .- Todo artículo estará relacionado con un usuario, excepto los procedentes del parseo de otras fuentes.

RS2 .- Todas las valoraciones estará relacionado con un usuario

RS3 .- El sistema de recomendación será un sistema de recomendación mixto

RS4 .- Todos los artículos serán públicos y no será necesario ser usuario para poder consultarlos.

4.1.2 Requisitos de rendimiento

El rendimiento de la plataforma debe ser ágil y no entorpecer la experiencia del usuario, por lo que las tareas con una carga alta al servidor se realizarán a horas en las que el número de usuarios sea mínimo, es decir a partir de las 2:00 hasta las 5:00. De este modo, las recomendaciones se genera sobre esa hora y se almacenarán todas las recomendaciones en base de datos.

4.2 *Modelos de caso de uso*

Para explicar de forma más clara y visual el comportamiento que tendrá el sistema usaremos un diagrama de casos de uso. De este modo podremos centrarnos en qué hace.

4.2.1 Diagramas de caso de uso

4.2.1.1 Actores

En este caso hablaremos de tan solo un actor, el usuario final. El usuario final será el único encargado de hacer llamadas a todas las funcionalidad de la plataforma, así como de aportarle información a la misma.

4.2.1.2 Diagrama

A continuación podemos ver el diagrama en el que se ve de forma visual y sencilla la forma en la que el usuario puede interactuar con la plataforma y el modo en el que puede llegar a todas sus funcionalidades.

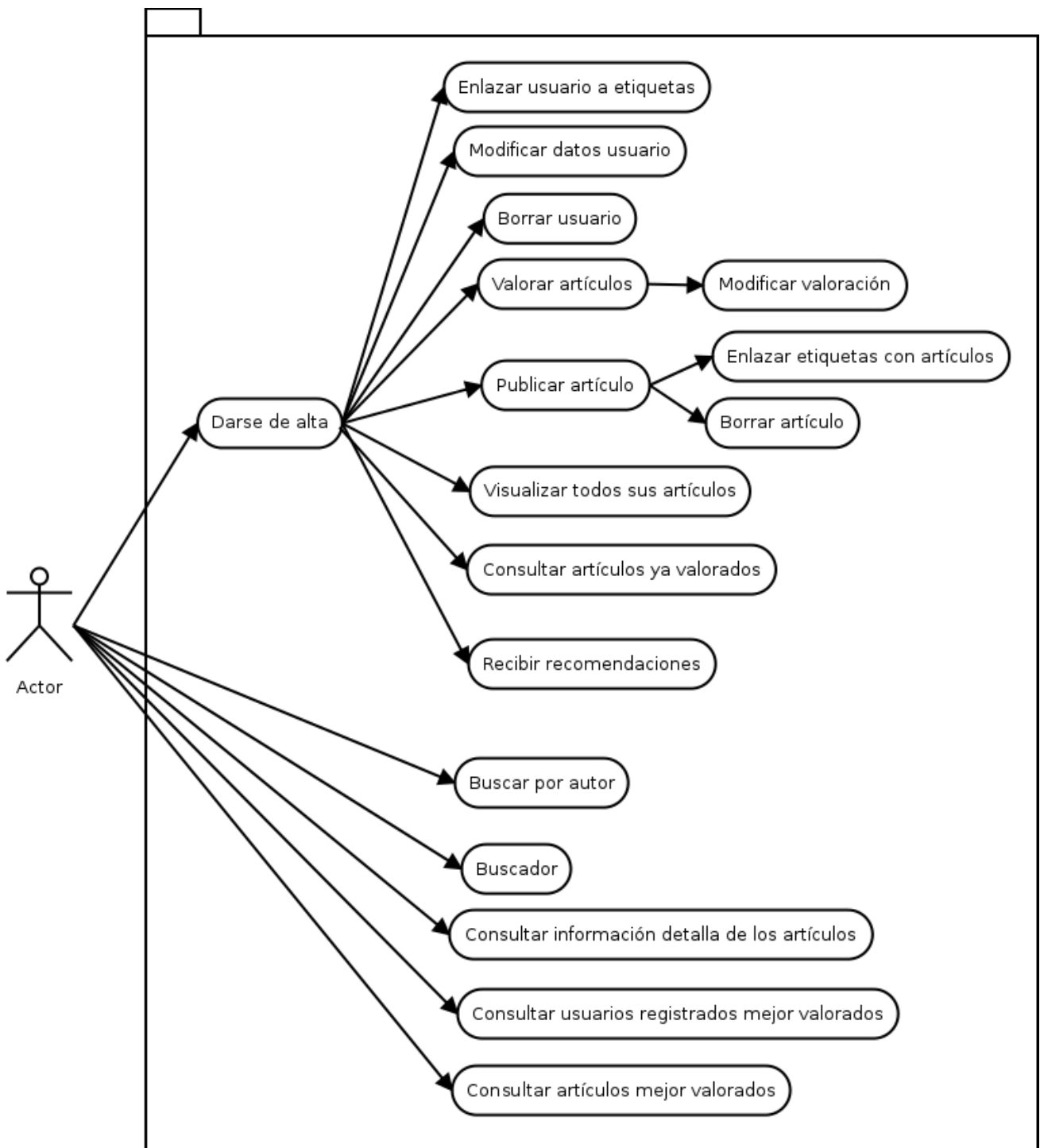


Imagen 1 - Diagrama de caso de uso

4.4 Esquemas de caso de uso

Para el correcto funcionamiento de la plataforma los usuarios deberán seguir los siguientes comportamientos en el uso de la aplicación.

- Alta usuario: Registra en base de datos los datos de dicho usuario.

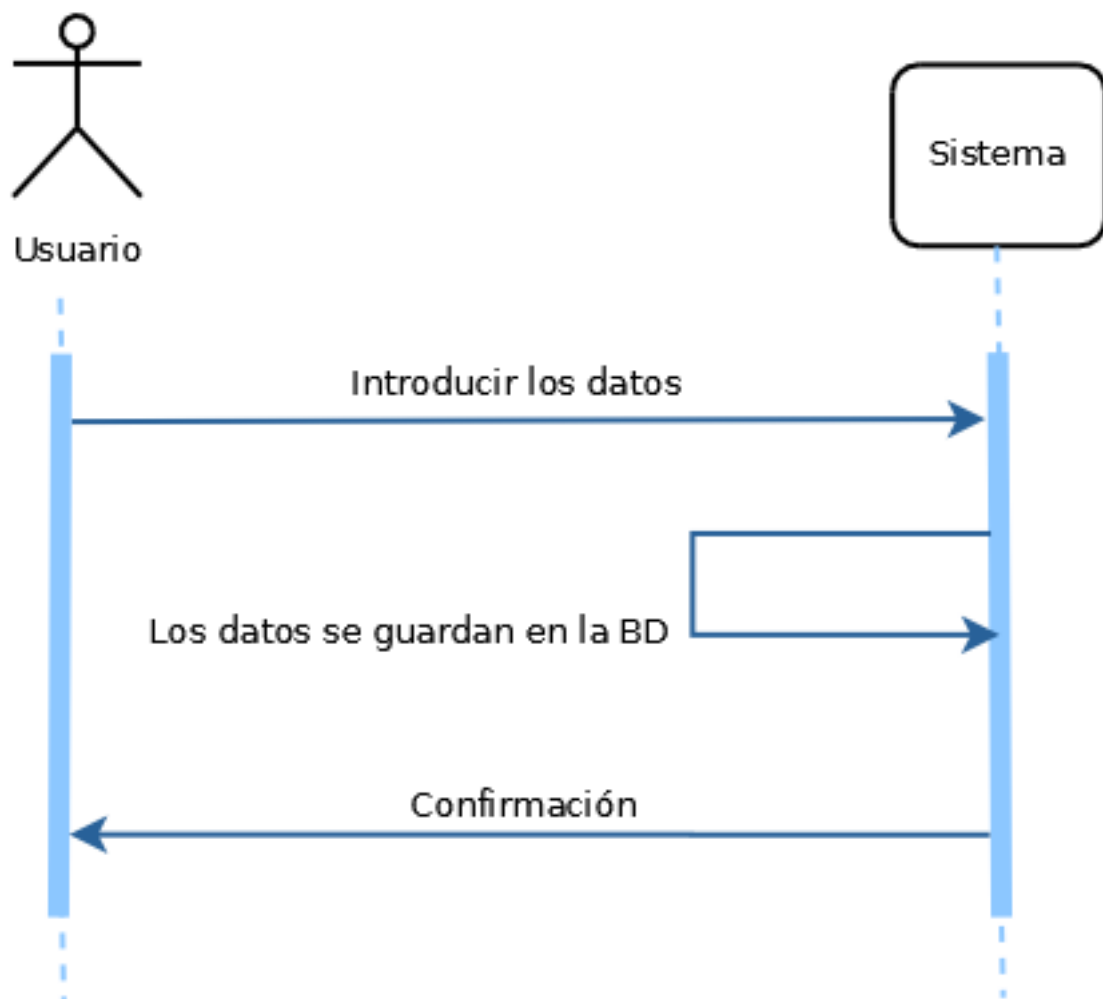


Imagen 2 - Diagrama de secuencia (Alta usuario)

- Modificar datos usuario: El usuario introduce los nuevos datos en su perfil y se actualizan en la base de datos.

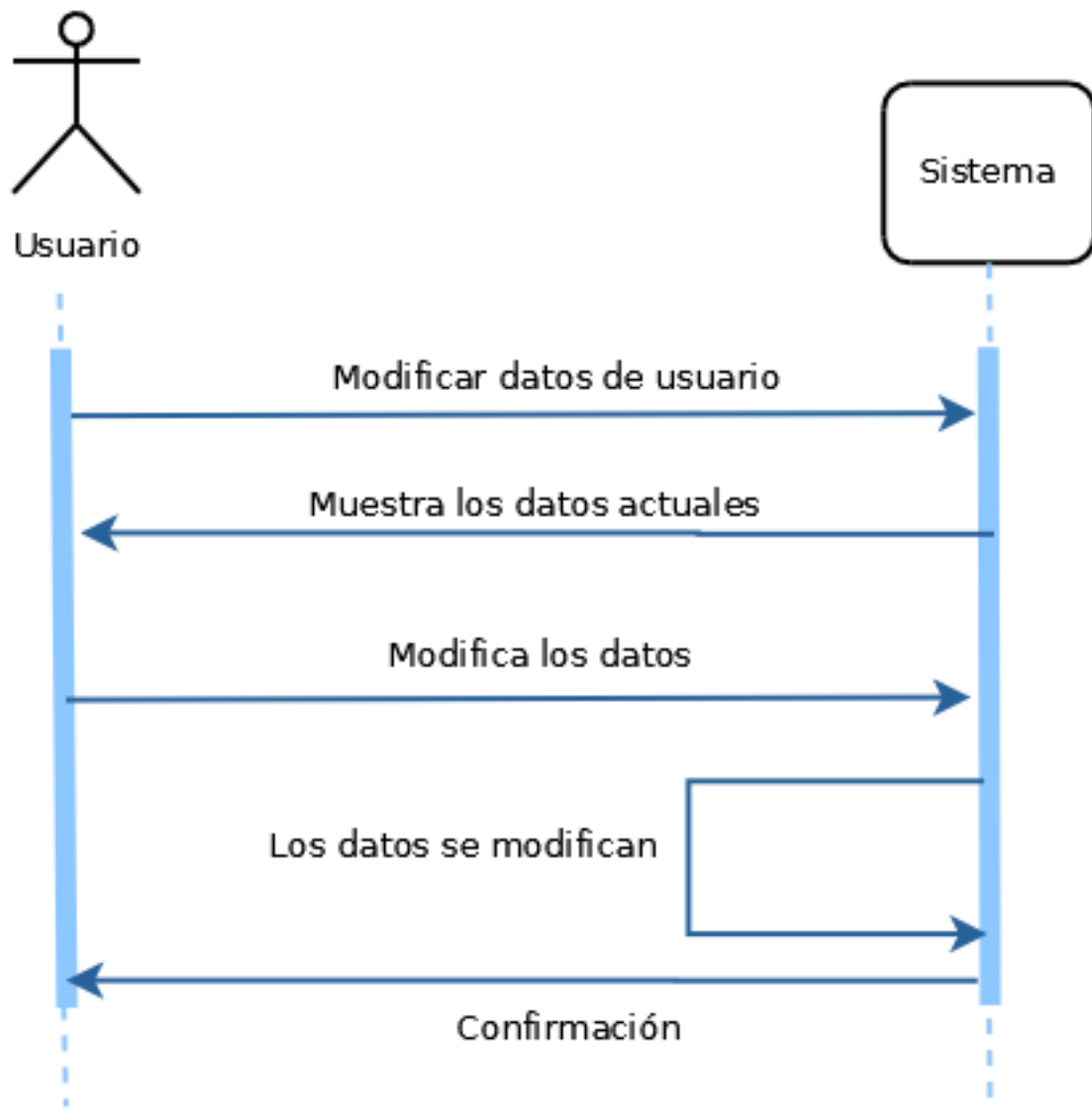


Imagen 3 - Diagrama de secuencia (Modificar datos usuario)

- Borrar usuario: El usuario borra su cuenta y toda la información que se tenía guardada de él.

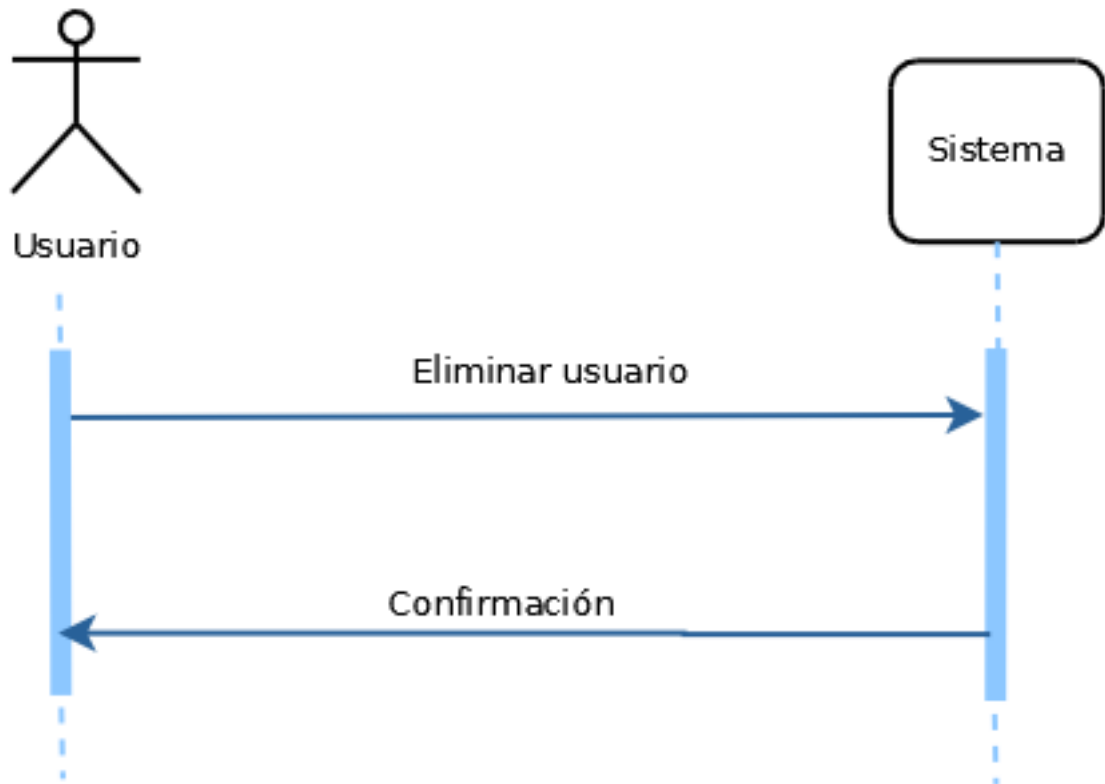


Imagen 4 - Diagrama de secuencia (Borrar usuario)

- **Publicar artículo:** Un usuario rellena los datos sobre su artículo y sube el fichero PDF en el que está el artículo. Además le asigna etiquetas descriptivas sobre el mismo. Se almacena todo en base de datos, se sube el fichero a un directorio del servidor y además se le indexa en el motor de búsqueda.

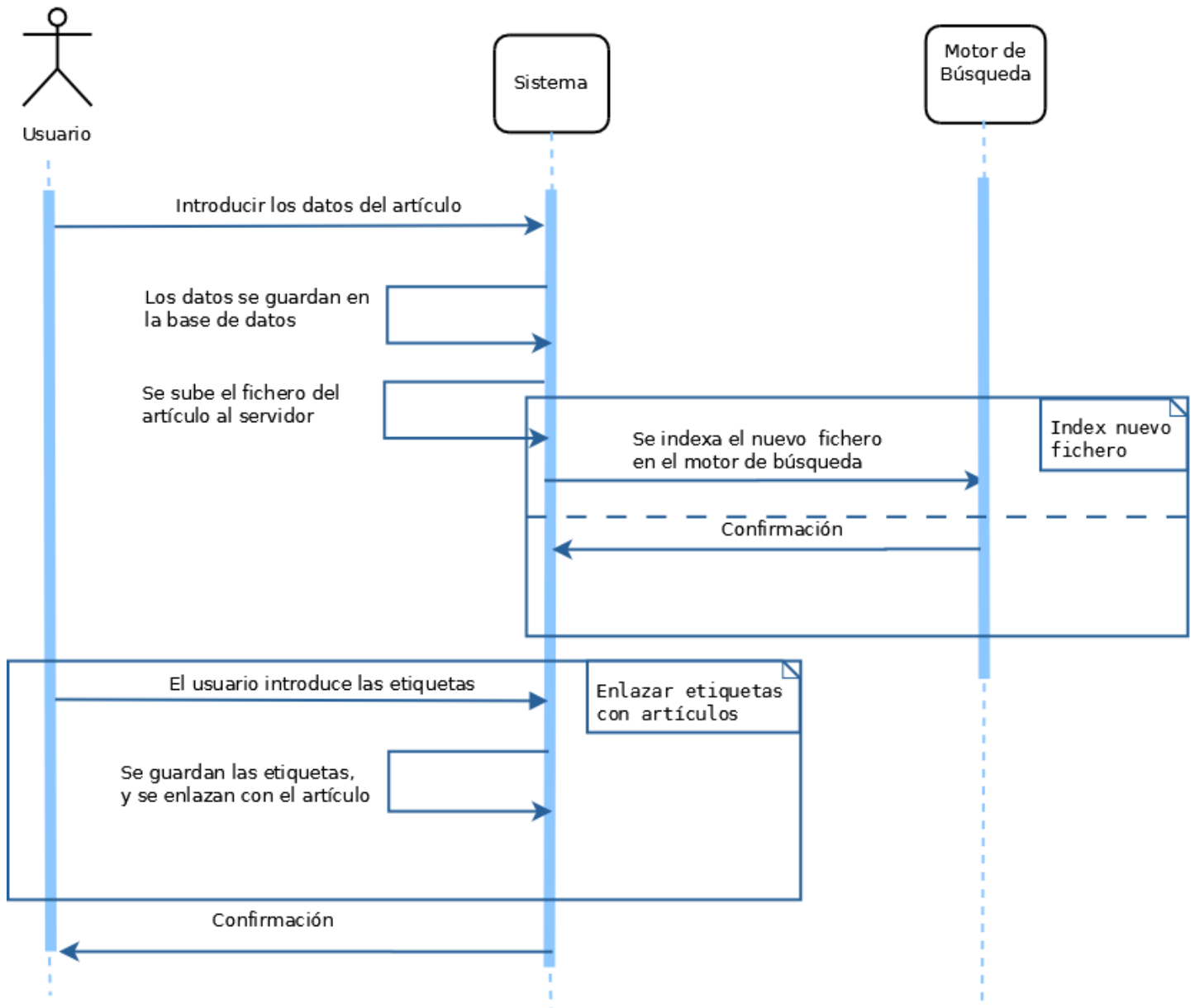


Imagen 5 - Diagrama de secuencia (Alta artículo)

- Borrar artículo: El usuario podrá borrar cualquiera de sus artículos, y toda la información relativa al artículo será borrada.

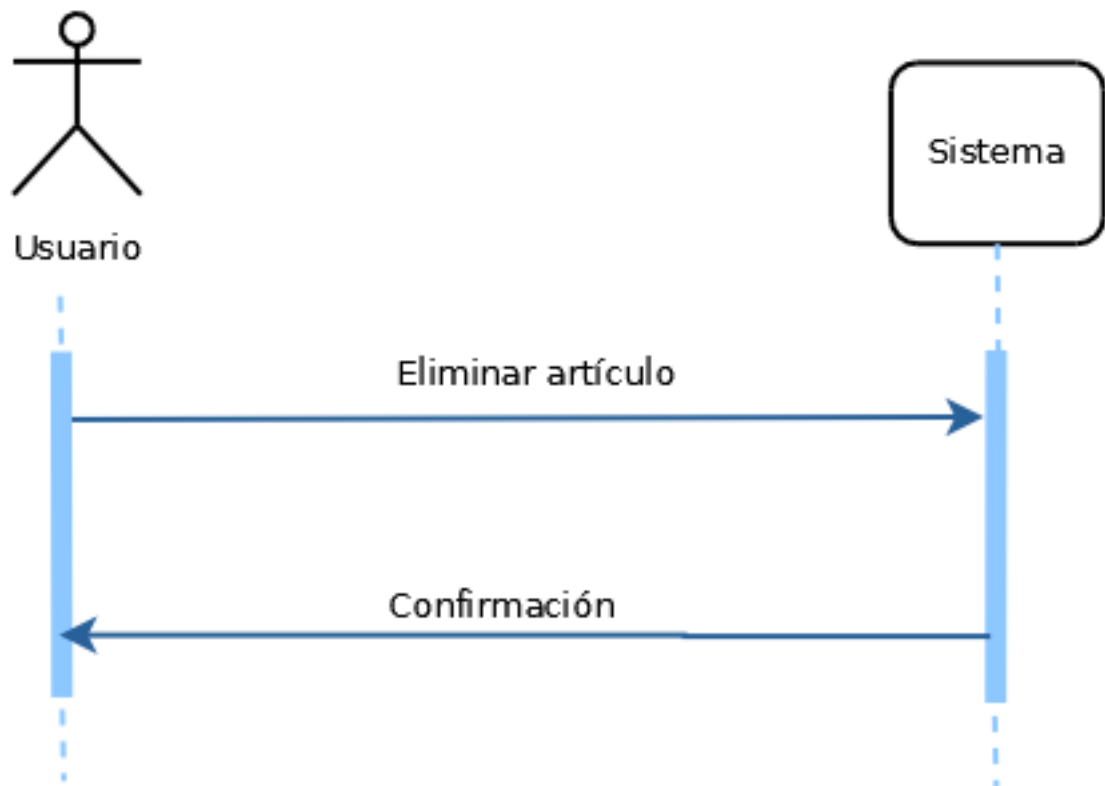


Imagen 6 - Diagrama de secuencia (Eliminar artículo)

- Buscador: Se introducirá en el buscador una palabra o cadenas de palabras, el motor de búsqueda realizará la búsqueda y devolverá los resultados a través de Json. La plataforma lo interpreta y los muestra correctamente.

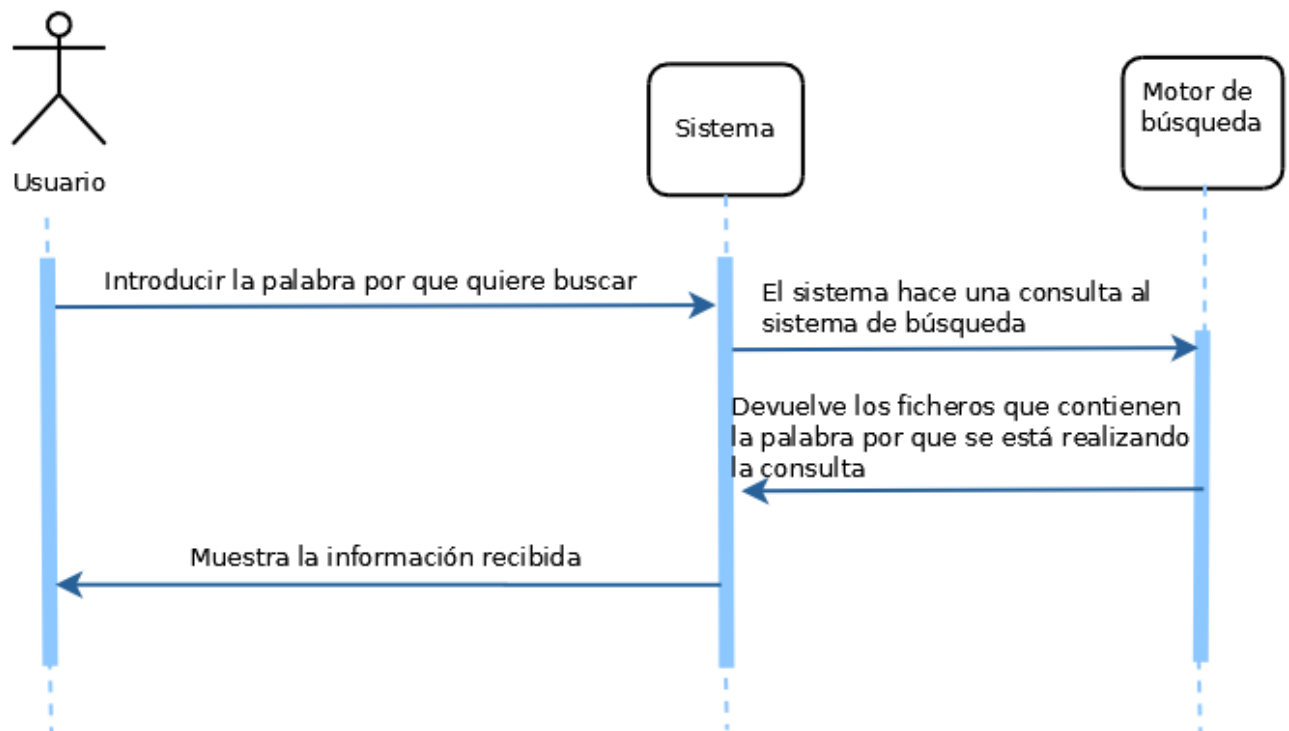


Imagen 7 - Diagrama de secuencia (Buscador)

- Buscador por autor: Se introduce el nombre del autor y la plataforma realiza una búsqueda en base de datos. Muestra los resultados.

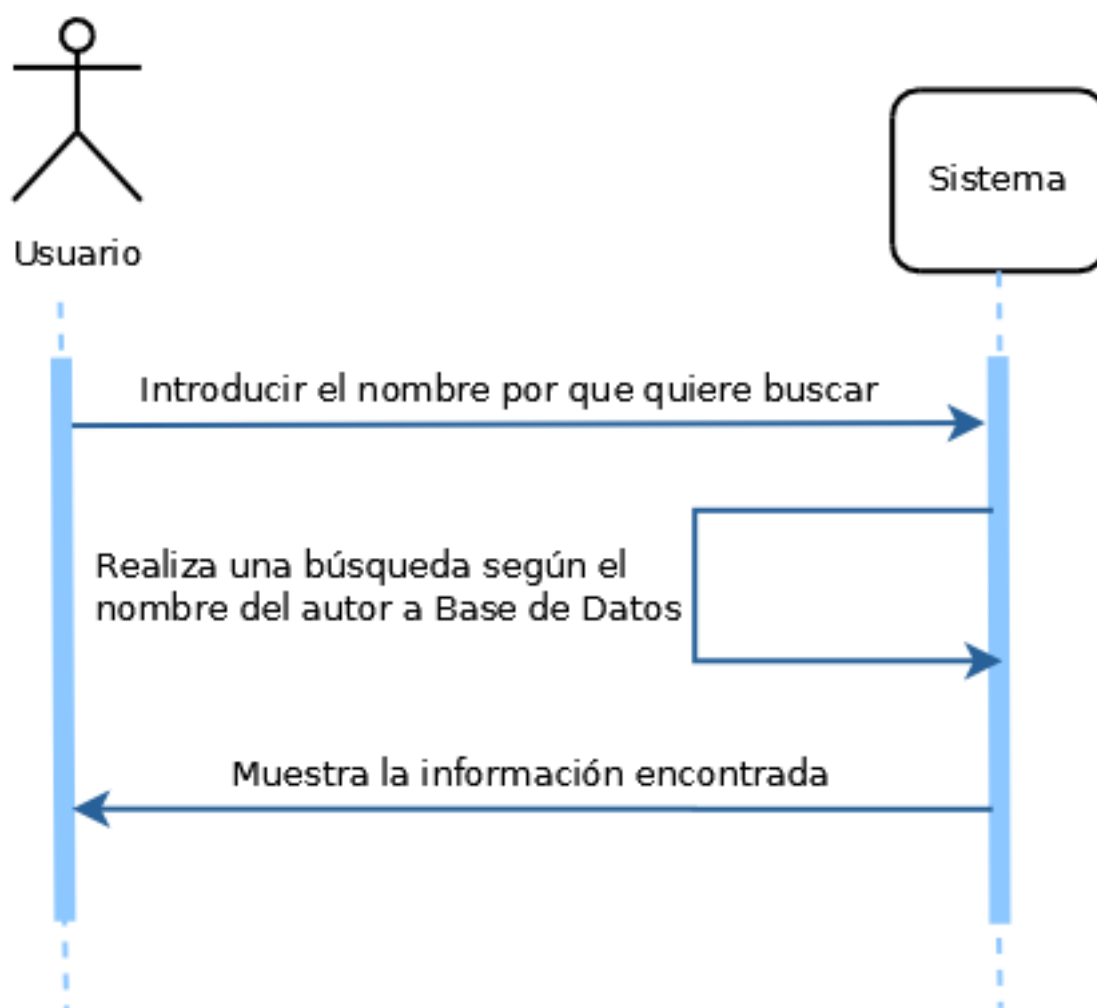


Imagen 8 - Diagrama de secuencia (Buscador por autor)

- Consultar información detallada del artículo: Al navegar entre los artículos el usuario podrá entrar en la información detallada del artículo y el PDF del artículo.

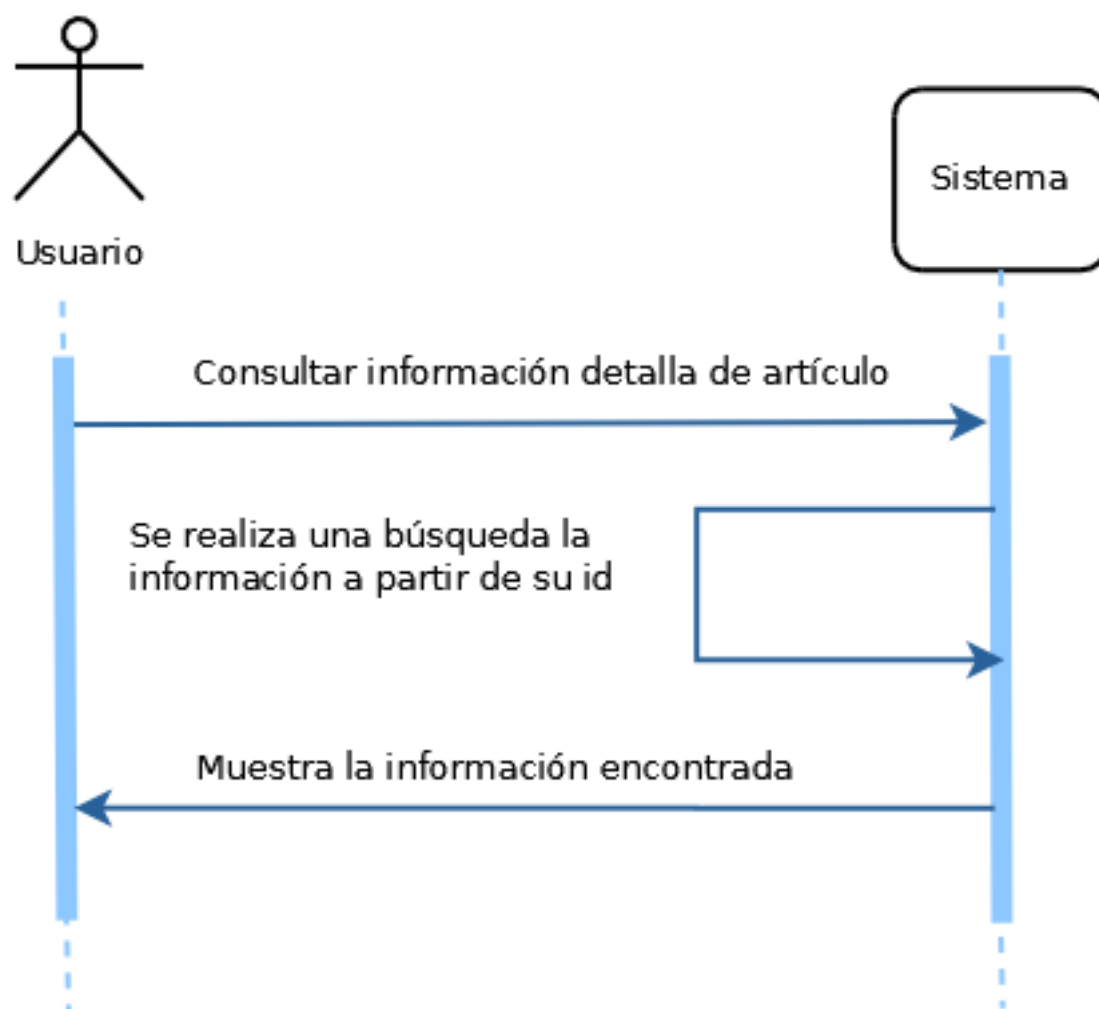


Imagen 9 - Diagrama de secuencia (Consultar detalles artículo)

- Valorar artículo: Al navegar por los artículos el usuario valorará con una puntuación numérica los artículos y se guardarán en base de datos.

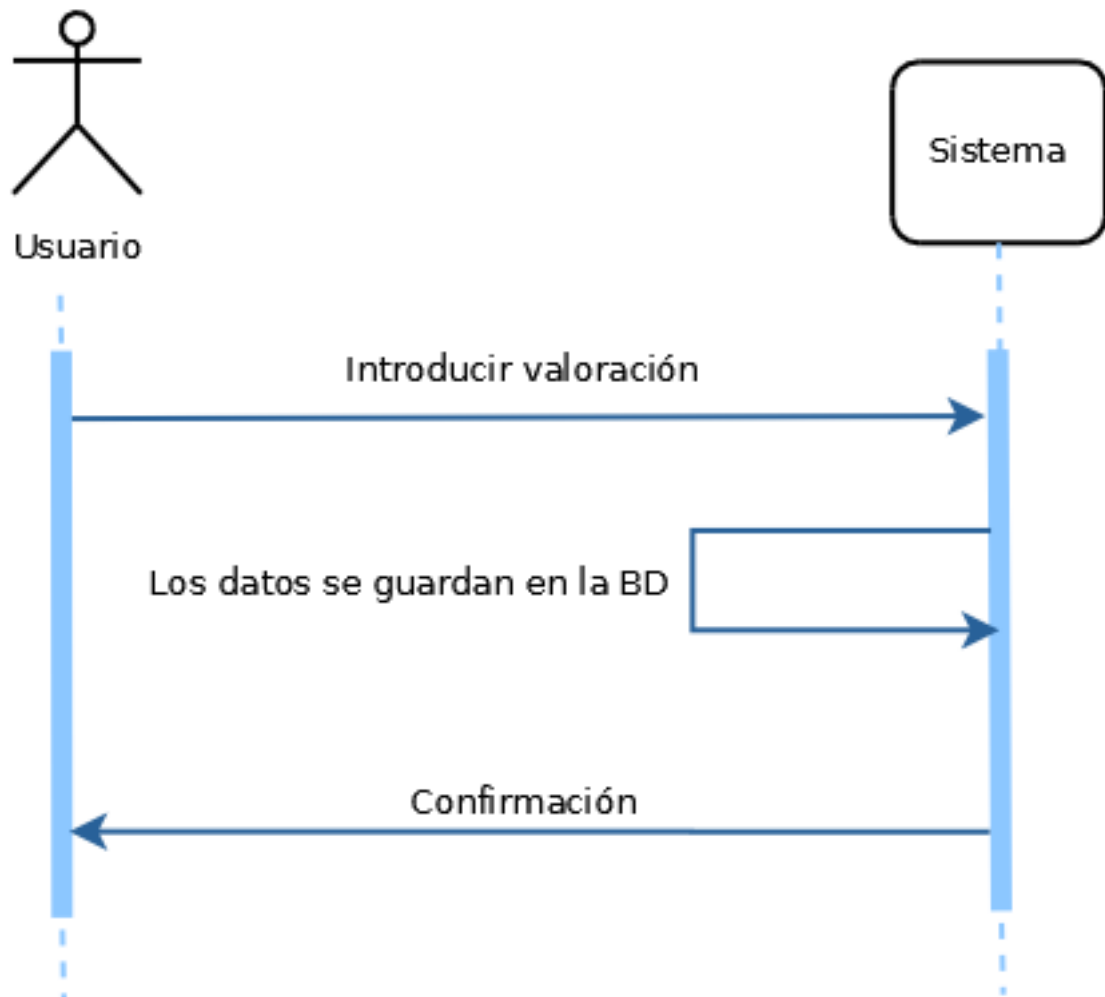


Imagen 10 - Diagrama de secuencia (Valorar artículo)

- Consultar artículos mejor valorados: El usuario accederá a la sección donde podrá ver el listado de artículos mejor valorados que habrá obtenido previamente el sistema al realizar la media de las valoraciones.

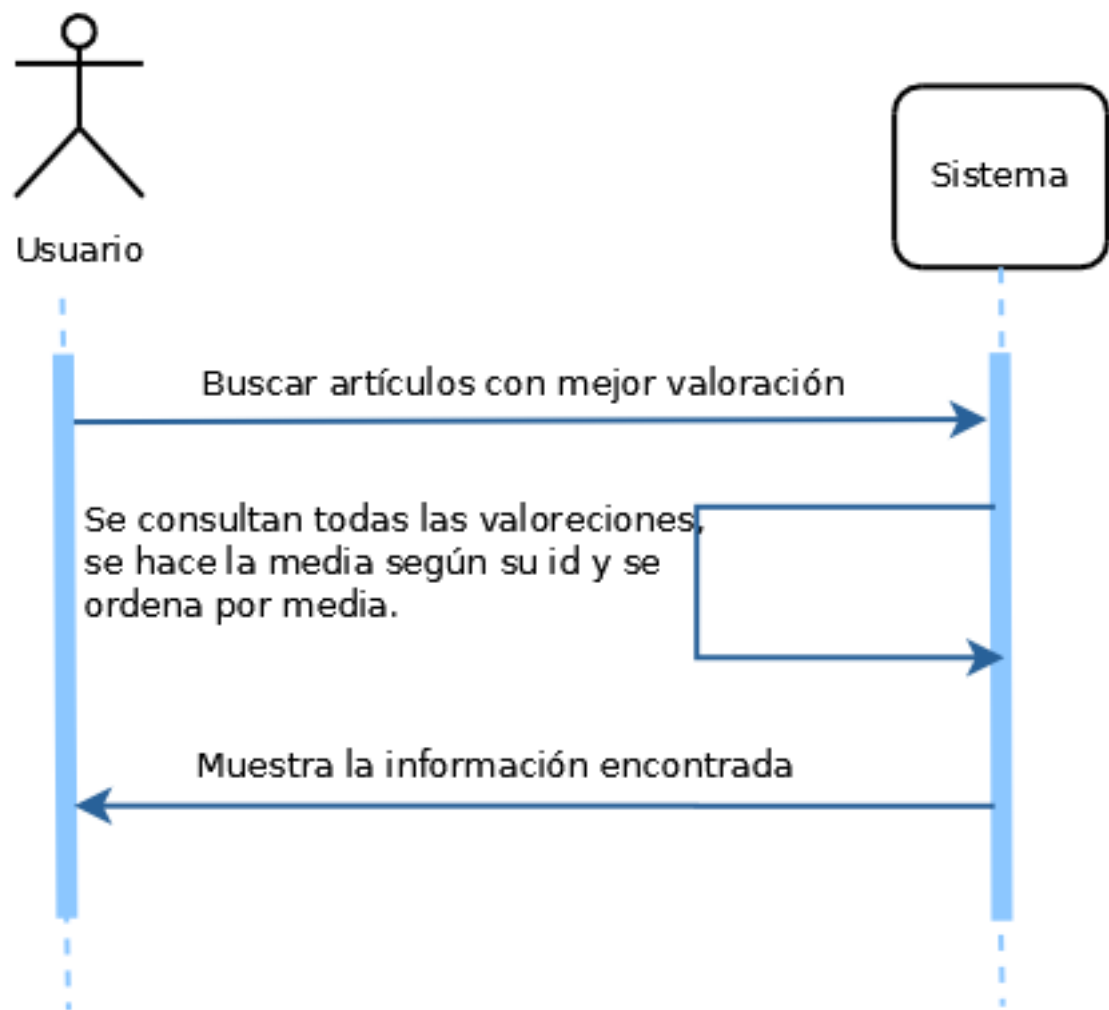


Imagen 11 - Diagrama de secuencia (Artículos mejor valorados)

- Consultar artículos ya valorados: El usuario accederá a la sección donde el sistema listará los artículos que el usuario ya ha valorado, así como las puntuaciones que realizó sobre los mismos.

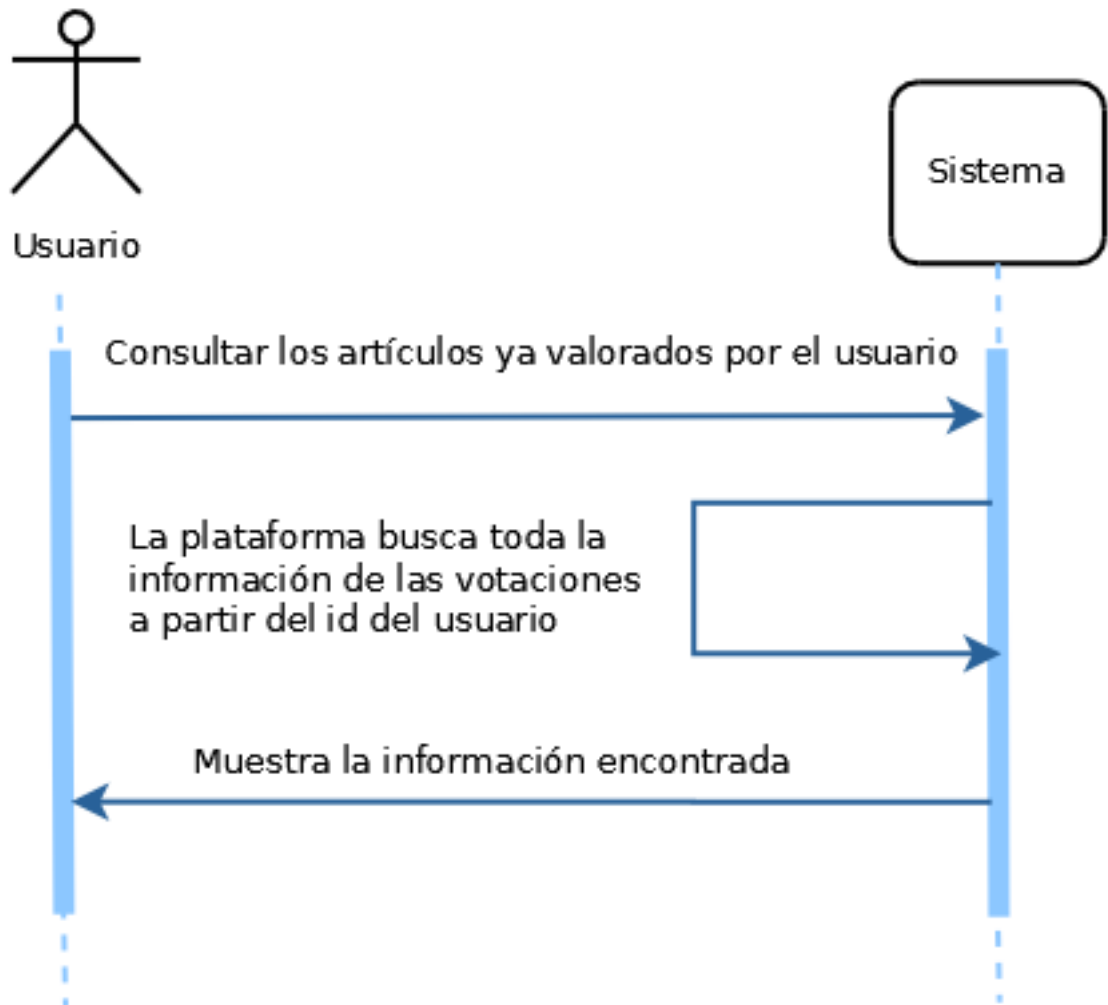


Imagen 12 - Diagrama de secuencia (consultar artículos valorados por el usuario)

- Consultar usuarios mejor valorados: Al acceder a esta sección de la plataforma el sistema listará los usuarios cuyo artículos han recibido las mejores valoraciones.

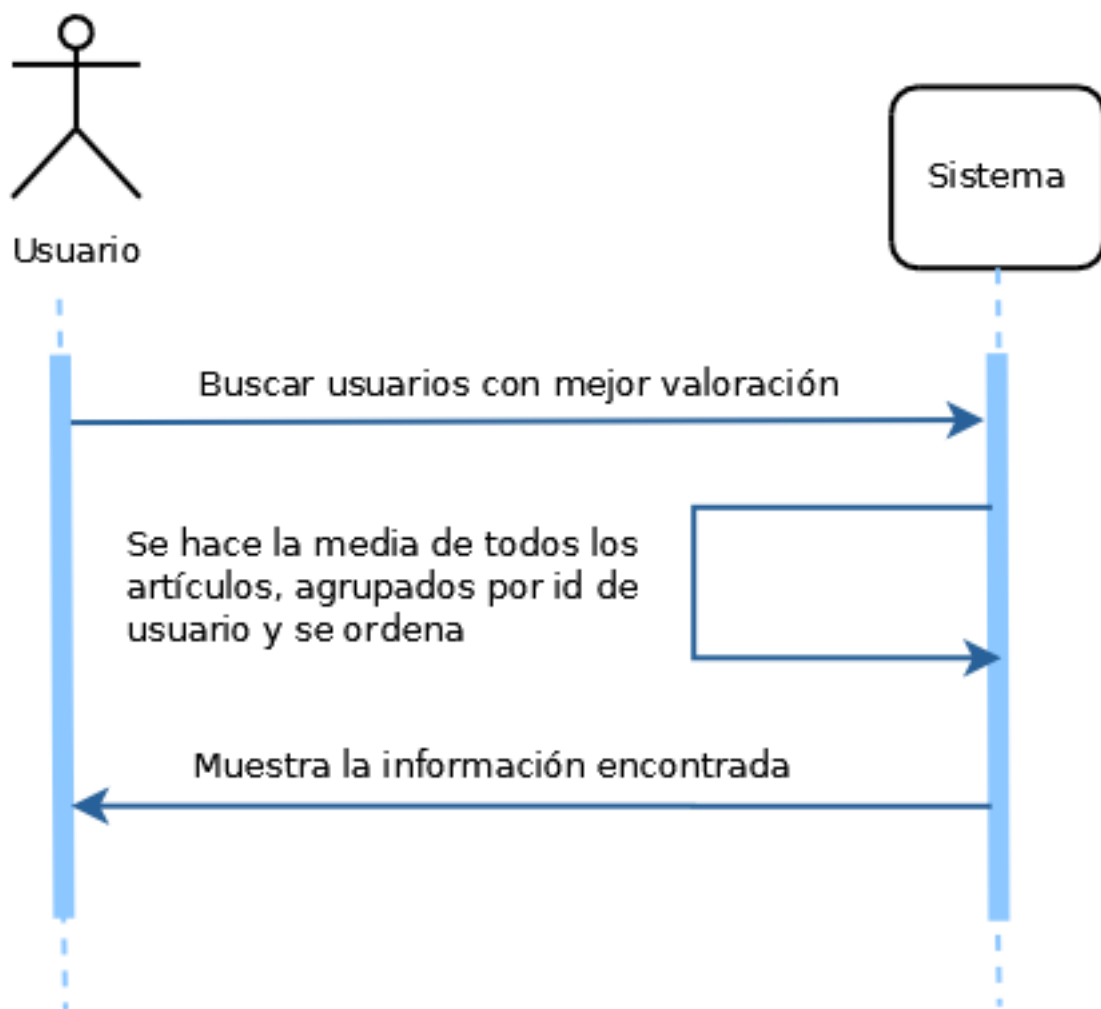


Imagen 13 - Diagrama de secuencia (usuarios mejor valorados)

- **Recomendar:** El sistema genera un archivo en el que se exportan las votaciones. A partir de este archivo el sistema de recomendación analiza los datos y obtiene recomendaciones a partir de las votaciones, relacionando los intereses de cada usuario a las votaciones similares. Una vez obtenidas las recomendaciones se insertan en base de datos. Cuando un usuario pide una recomendación se consulta de la base de datos.

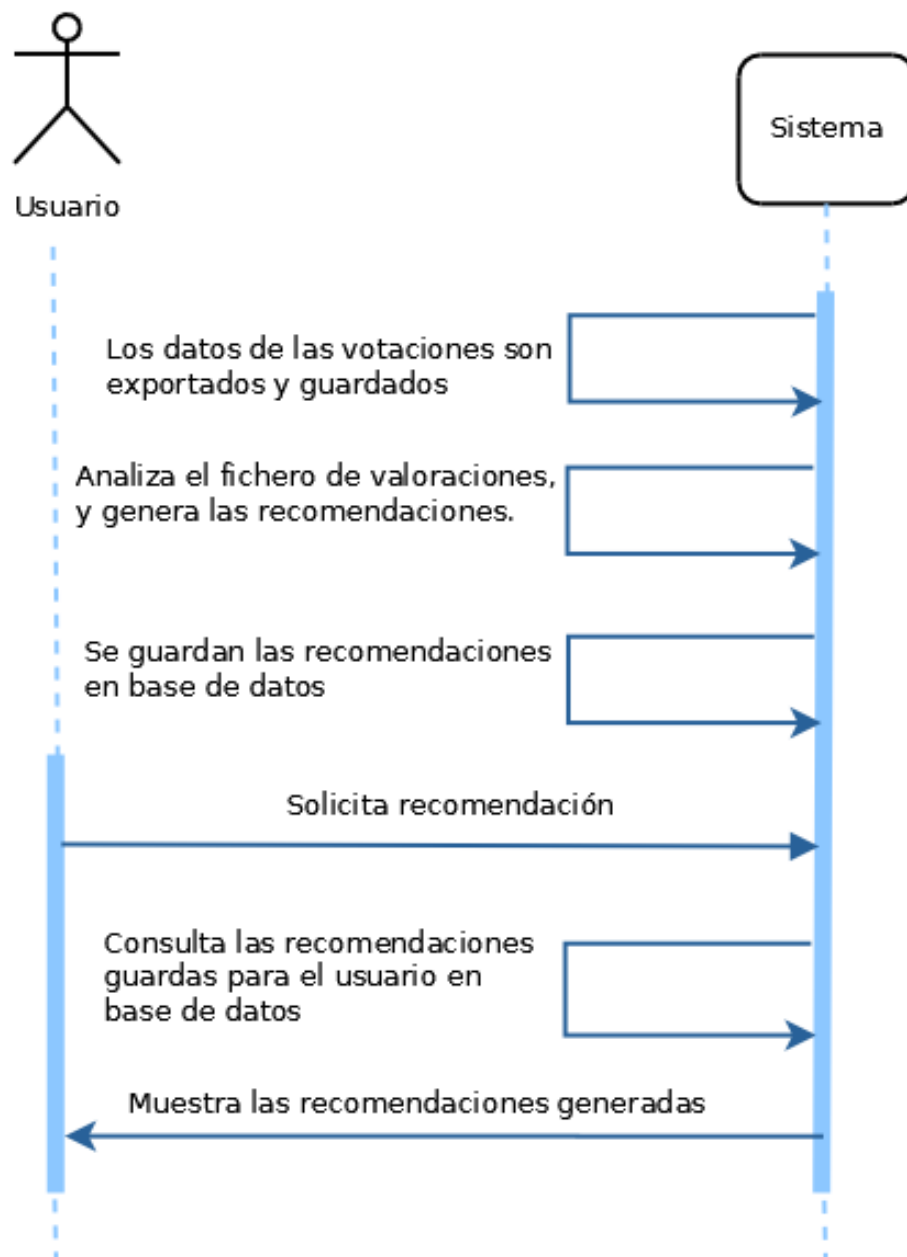


Imagen 14 - Diagrama de secuencia (Recomendaciones)

5. Diseño

5.1 Implementación y desarrollo

Tras analizar la plataforma y todas las funcionalidad que lleva con ella se eligieron los lenguajes de programación que se iban a usar y se llevo a la conclusión de que se desarrollaría con los siguientes paradigmas de programación:

- Plataforma web: llevada a cabo una programación estructura[8], con el fin de agilizar el desarrollo y de hacer las tareas de depuración y corrección de errores más sencilla.
- Sistema de recomendación: En este caso al estar desarrollado en Java era claro que debe seguir el paradigma de programación orientada a objetos[9].

5.2 Diseño de base de datos

Este diagrama nos muestra la relación que guardan las entidades, es decir, la mera en la que interactúan cada una de las entidades que forman nuestro proyecto. A partir de este esquema podemos obtener todas las tablas que va a tener la plataforma, así como las tablas que marcaran la relación de tipo 1...n bidireccional.

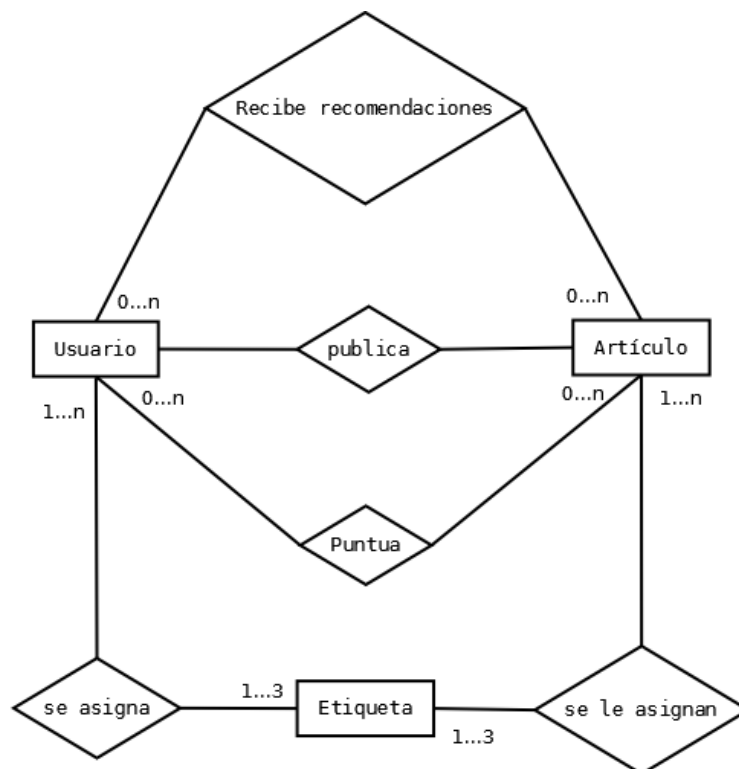


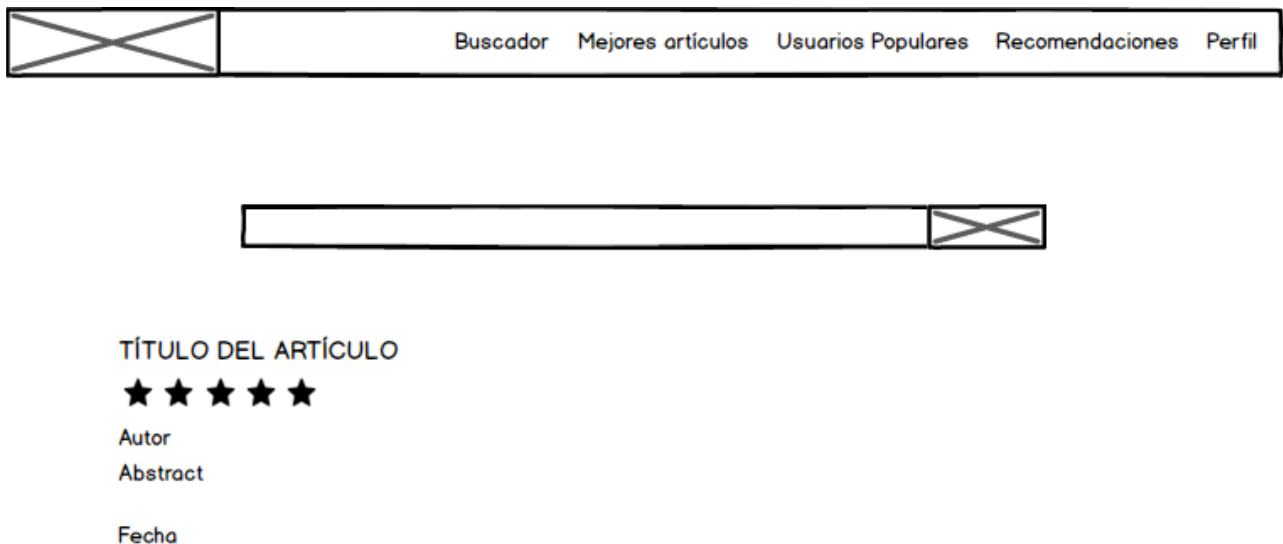
Imagen 15 - Esquema Entidad-Relación

5.3 Diseño de la plataforma

Para facilitar la maquetación de la plataforma se diseñaron inicialmente mockups, en los que se intentó conseguir un estilo sencillo y funcional, el que las llamadas a la acción y los puntos de interés estuviesen resultados con diferentes colores.

5.3.1 Buscador

En esta página se tendrá un input para realizar la búsqueda y se listarán los resultados.



The mockup illustrates the layout of a search page. At the top, there is a horizontal navigation bar with a logo on the left and five links: 'Buscador', 'Mejores artículos', 'Usuarios Populares', 'Recomendaciones', and 'Perfil'. Below this bar is a large, empty rectangular area intended for search results. Underneath the results area, a sample article entry is shown, consisting of the article title, a five-star rating, the author's name, the abstract text, and the publication date.

TÍTULO DEL ARTÍCULO

★ ★ ★ ★ ★

Autor

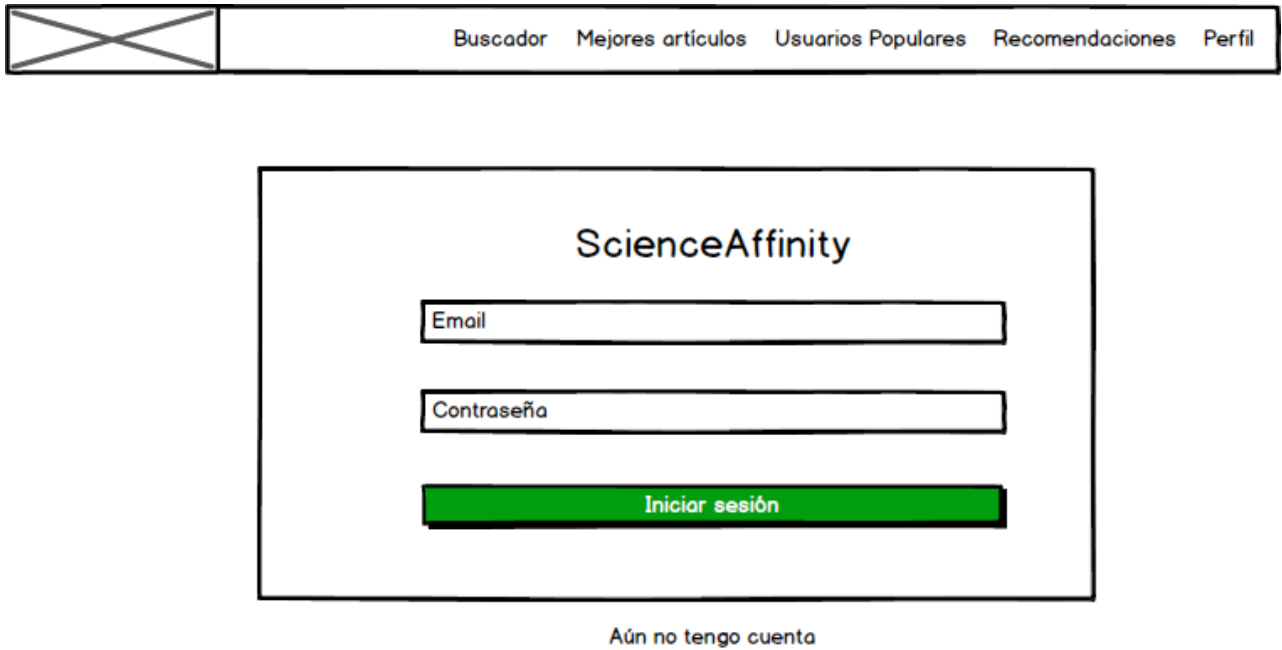
Abstract

Fecha

Imagen 16 - Diseño pantalla buscador

5.3.2 Iniciar sesión

En esta página se encontrará un formulario de login y además un enlace para que los usuarios que no tengan cuenta se puedan dar de alta en la plataforma.



The image shows a login interface for 'ScienceAffinity'. At the top, there is a navigation bar with a placeholder icon (a box with an 'X') on the left and a list of links on the right: 'Buscador', 'Mejores artículos', 'Usuarios Populares', 'Recomendaciones', and 'Perfil'. Below the navigation bar is a central login box. Inside this box, the title 'ScienceAffinity' is centered at the top. Below the title are two input fields: the first is labeled 'Email' and the second is labeled 'Contraseña'. Below these fields is a green button with the text 'Iniciar sesión'. At the bottom of the login box, there is a link that says 'Aún no tengo cuenta'.

Imagen 17 - Diseño pantalla de inicio de sesión

5.3.3 Crear cuenta

En esta sección de la plataforma aparece un formulario sencillo, con el fin de que no sea demasiado pesado para el usuario, en el que introducirán datos básicos para crear la cuenta. Además este diseño será aprovechado para la sección en la que el usuario puede modificar sus datos, pues es un formulario con el que el usuario está familiarizado.

	Buscador Mejores artículos Usuarios Populares Recomendaciones Perfil
---	--

Datos personales

Nombre

Apellidos

Puesto de trabajo

Organización en la que trabaja

Titulación

Datos de acceso

Email

Contraseña

Tus intereses

Campo de trabajo

Tema principal de interés

Tema última investigación

Crear cuenta

Imagen 18 - Diseño pantalla de registro

5.3.4 Información completa del artículo

En esta sección de la plataforma vamos a encontrar toda la información de cada artículo y podremos realizar las votaciones. Además encontramos un enlace en el que podemos visualizar el artículo completo en PDF.

BuscadorMejores artículosUsuarios PopularesRecomendacionesPerfil

Título

AutorFecha publicación

Abstract

Texto del abstract

Votaciones

Puntuación

★ ★ ★ ★ ★

Puntúalo

Listado puntos▼

Artículo

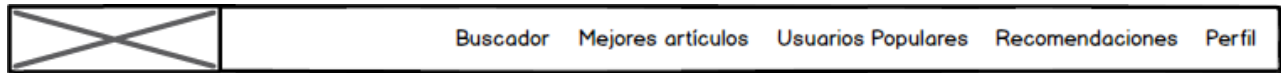
Leer artículo completo

>

Imagen 19 - Diseño de la pantalla de información sobre artículos

5.3.5 Usuarios ya valorados por el usuario

En esta página se encuentran los artículos que ya ha votado el usuario en un listado con la votación que le ha dado.



Los artículos que ya has valorado

TÍTULO DEL ARTÍCULO

★ ★ ★ ★ ★

Tu puntuación: ★ ★ ★ ★ ★

Autor

Abstract

Fecha

TÍTULO DEL ARTÍCULO

★ ★ ★ ★ ★

Tu puntuación: ★ ★ ★ ★ ★

Autor

Abstract

Fecha

Imagen 20 - Diseño de la pantalla de artículos ya valorados

5.3.6 Artículos del usuario

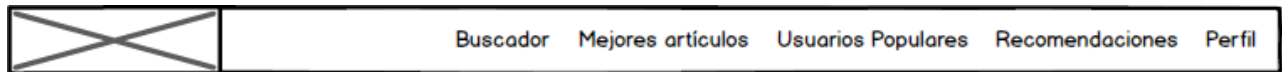
En esta página se encuentran los artículos que el usuario ha publicado en la plataforma.



Imagen 21 - Diseño de la pantalla de artículos ya publicados por el usuario

5.3.7 Artículos mejor valorados

En este apartado se encuentran los 20 artículos que mejor valoración han obtenido por parte los usuarios.



Top 20 mejores artículos

TÍTULO DEL ARTÍCULO



Autor

Abstract

Fecha

TÍTULO DEL ARTÍCULO



Autor

Abstract

Fecha

TÍTULO DEL ARTÍCULO



Autor

Abstract

Fecha

Imagen 21 - Diseño de la pantalla de mejores artículos

5.3.8 Usuarios mejor valorados

En este apartado se encuentran los 20 usuarios cuya media de todas las valoraciones que han recibido sus artículos es más alta.


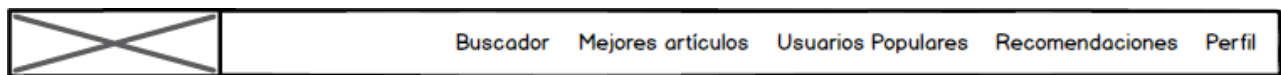
		Buscador	Mejores artículos	Usuarios Populares	Recomendaciones	Perfil
Top 20 usuarios mejor valorados						
1	Nombre Organización	Media: 0.00 En n artículos				
2	Nombre Organización	Media: 0.00 En n artículos				
3	Nombre Organización	Media: 0.00 En n artículos				

Imagen 22 - Diseño de la pantalla de usuarios mejor valorados

5.3.9 Recomendaciones

La plataforma divide las recomendaciones en dos secciones las recomendaciones, una sección para el filtrado por contenido y otro por filtrado colaborativo. Ambas secciones compartirán diseño, la única diferencia será un pequeño texto explicativo y el título.



Recomendaciones para tí

Breve explicación sobre estas recomendaciones

TÍTULO DEL ARTÍCULO



Autor

Abstract

Fecha

TÍTULO DEL ARTÍCULO



Autor

Abstract

Fecha

TÍTULO DEL ARTÍCULO



Autor

Abstract

Fecha

Imagen 23 - Diseño de la pantalla de artículos ya valorados

6. Arquitectura

6.1 Arquitectura del sistema

En esta sección se puede encontrar una explicación breve sobre cada parte de la estructura del proyecto, donde podremos encontrar desde la estructura de datos de la misma hasta la estructuración de los ficheros del desarrollo. En cada sección podremos encontrar una explicación del uso de cada tecnología y de la estructura seguida.

Para empezar y tener una visión más general de la plataforma el siguiente diagrama nos muestra la relación entre los módulos y de estos módulos con sus librerías.

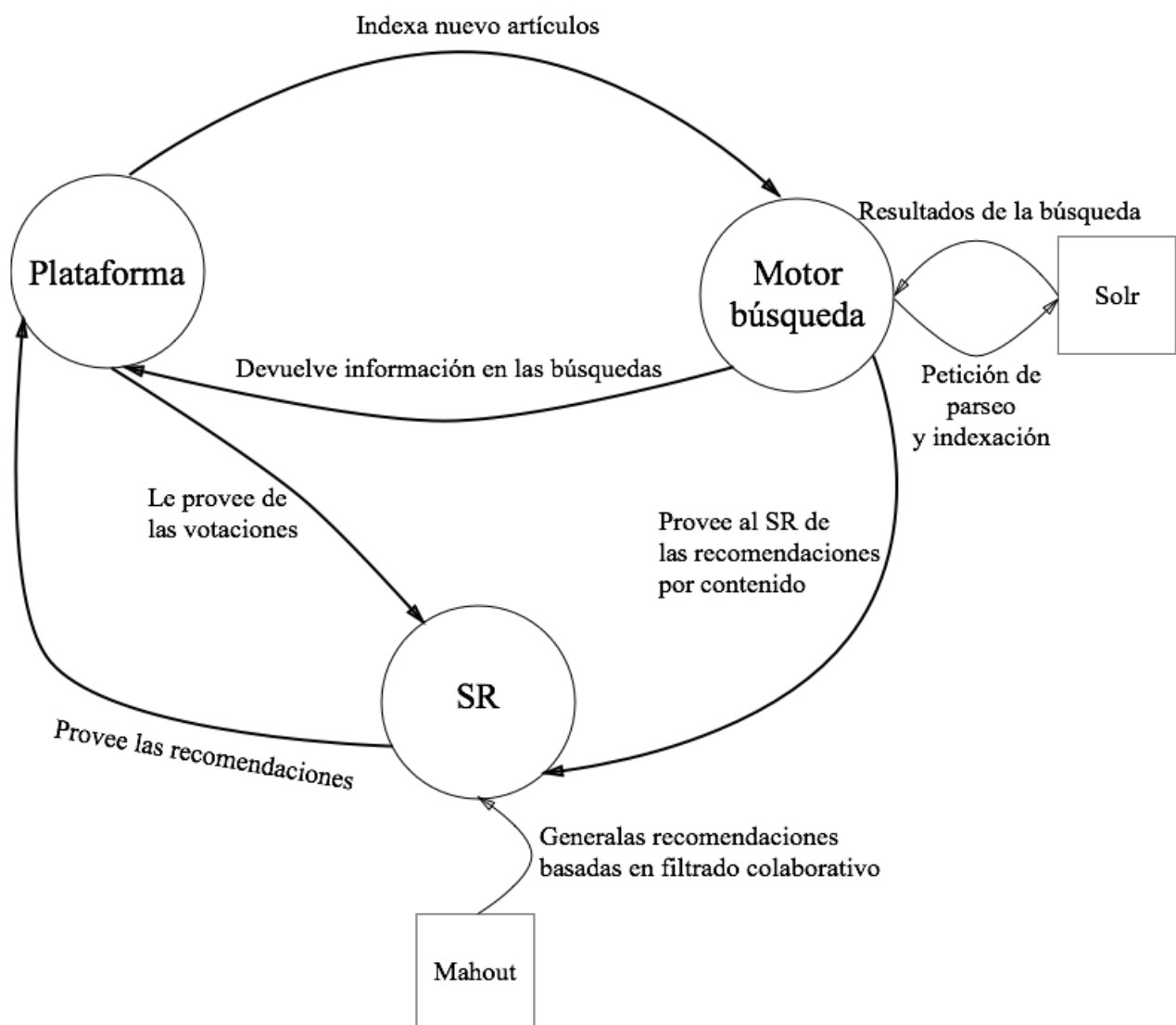
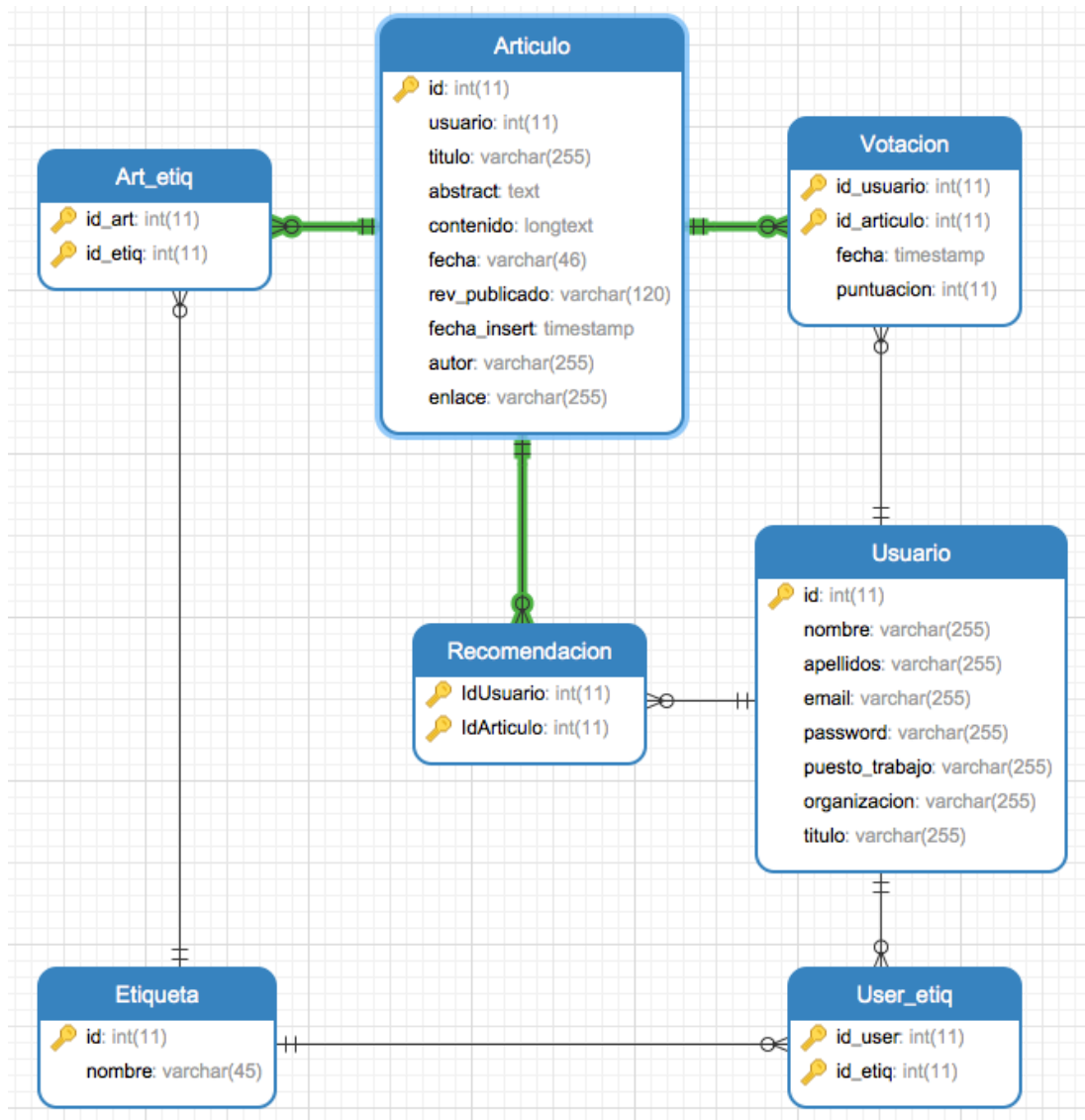


Imagen 24 - Esquema relación entre módulos

6.1.1 Base de datos

La base de datos es una base de datos MySQL, que sigue la siguiente estructura:



A partir de esta estructura, podremos encontrar los siguientes datos en cada tabla:

Tabla Usuario

Columns:	
<u>id</u>	int(11) AI PK
nombre	varchar(255)
apellidos	varchar(255)
email	varchar(255)
password	varchar(255)
puesto_trabajo	varchar(255)
organizacion	varchar(255)
titulo	varchar(255)

Tabla Artículo

Columns:	
<u>id</u>	int(11) AI PK
usuario	int(11)
titulo	varchar(255)
abstract	text
contenido	longtext
fecha	varchar(46)
rev_publicado	varchar(120)
fecha_insert	timestamp
autor	varchar(255)
enlace	varchar(255)

Tabla Etiqueta

Columns:	
<u>id</u>	int(11) AI PK
nombre	varchar(45)

Tabla Recomendación

Columns:

<u>IdUsuario</u>	int(11) PK
<u>IdArticulo</u>	int(11) PK

Tabla Votación

Columns:

<u>id_usuario</u>	int(11) PK
<u>id_articulo</u>	int(11) PK
fecha	timestamp
puntuacion	int(11)

Tabla User_Etiq

Columns:

<u>id_user</u>	int(11) PK
<u>id_etiq</u>	int(11) PK

Tabla Art_Etiq

Columns:

<u>id_art</u>	int(11) PK
<u>id_etiq</u>	int(11) PK

De este modo se cumplen todas las necesidades en cuanto a datos se refiere. Se ha usado una estructura lo más simple y funcional posible.

6.1.2 Motor de búsqueda

En cuanto al sistema de búsqueda se usa Hadoop, para analizar gramaticalmente todos los archivos PDF y poder hacer búsquedas internas dentro del archivo.

Dentro de Hadoop encontramos colecciones que guardan toda la información de los archivos guardados y las cuales permiten realizar búsquedas gramaticales dentro de los ficheros.

6.1.3 Sistema de recomendación

El sistema de recomendación esta creado a partir de Mahout, herramienta de machine learning, la cual nos provee de las herramientas necesarias para la creación del vecindario y las correspondientes recomendaciones.

Es una de las partes principales del proyecto y se encontrará apartada de los ficheros web de la plataforma, pues al estar desarrollado en Java el fichero ejecutable estará en otra carpeta no accesible para los usuarios donde se ejecutará periódicamente.

6.1.4 Secciones de la plataforma

Al entrar en a la plataforma por primera vez, si tener un usuario registrado, se divide en dos partes principales:

- Buscador
- Alta de usuario
- Artículos mejor valorados
- Usuarios mejor valorados

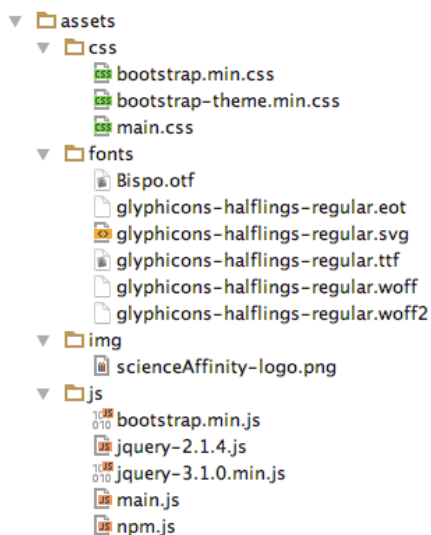
Una vez que el usuario esta registrado la parte privada se divide en:

- Buscador
- Mis recomendaciones, dividido en:
 - Basadas en item
 - Basadas en vecindario
- Mi perfil
- Artículos mejor valorados
- Usuario mejor valorados
- Mis artículos, donde además encontramos:
 - Añadir nuevo articulo

6.1.5 Distribución de ficheros

Dentro del proyecto los archivos han sido divididos en varios directorios, dependiendo de las funciones de estos ficheros. Se explican a continuación:

A partir del directorio raíz encontramos las siguientes sub-carpetas:



- **Assets:** En esta carpeta podemos encontrar los ficheros auxiliares que contienen temas de estilos, fuentes, imágenes y JavaScript.

Lo más relevante en esta carpeta puede ser la inclusión de Bootstrap, como framework de CSS incorporado para ayudar en la tarea de hacer la web responsiva.

Imagen 25 - Distribución de archivos en carpeta 'Assets'

- **Aux:** En esta carpeta podemos encontrar ficheros php a los que o bien se llama desde algún formulario o bien realizan una función auxiliar independiente de la interacción del usuario.

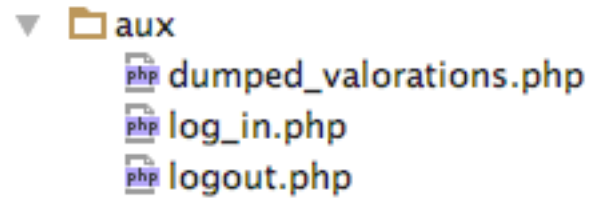


Imagen 26 - Distribución de archivos en carpeta 'aux'

- **Include:** En este directorio podemos encontrar ficheros que se irán incluidos a la mayoría de las vistas de la plataforma web. Los más destacables son los ficheros functions.php que contiene todas las funciones PHP desarrolladas (conexiones a bases de datos, recoger información de base de datos, subir ficheros, etc) y el fichero config.php (en el que se configura la conexión a base de datos)

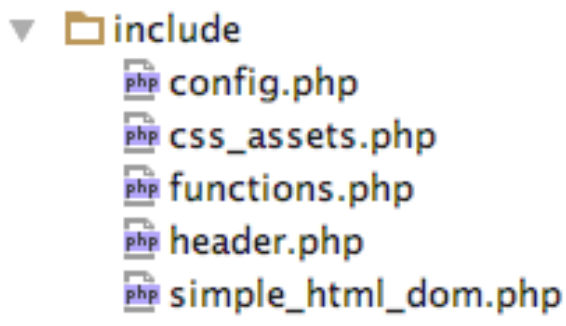


Imagen 27 - Distribución de archivos en carpeta 'include'

- **Solr:** Aquí se encuentran todos los ficheros del motor de búsqueda Solr, el encargado de parsear e indexar todos los artículos.

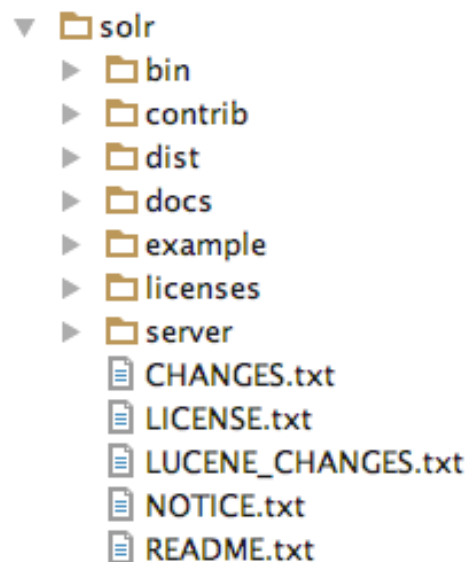
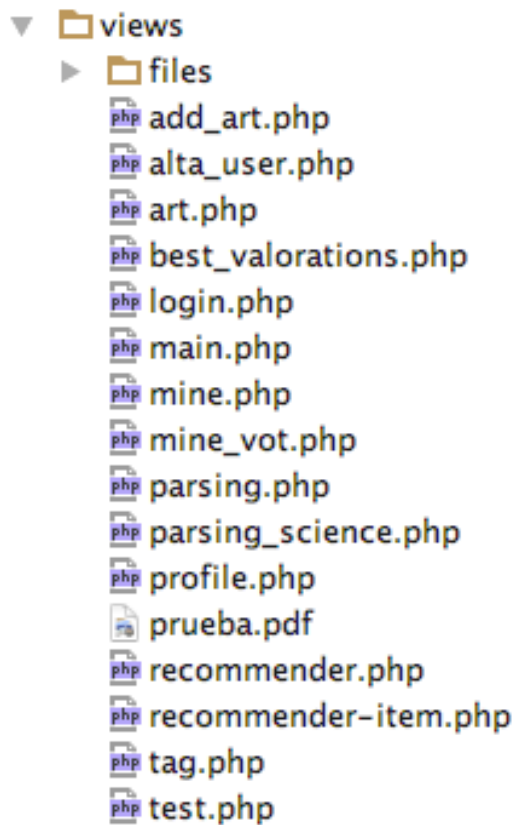


Imagen 28 - Distribución de archivos en carpeta 'solr'



- **Views:** Dentro de este directorio encontramos todos los ficheros de las vistas de la plataforma, es decir, de todas las páginas que forman la plataforma web.

Imagen 29 - Distribución de archivos en carpeta 'views'

7. Desarrollo

7.1 Herramientas utilizadas

A lo largo del proceso de desarrollo se ha hecho uso de varias tecnologías, como ya hemos mencionado anteriormente. En este capítulo vamos a encontrar qué son esas tecnologías, el uso que se le ha dado y por qué ha sido seleccionada cada tecnología.

7.1.1 Lenguajes de programación

En este apartado vamos a encontrar la descripción, características y en qué secciones se ha utilizado cada uno de ellos.

7.1.1.1 PHP

PHP (Pre Hypertext -processor) es un lenguaje de programación orientado principalmente al entorno web. Fue creado por Rasmus Lerdorf y apareció en el año 1995. Actualmente se encarga de su mantenimiento y desarrollo The PHP Group[10].

La principal característica a remarcar de este lenguaje de programación es que genera código del lado del servidor, es decir, toda la carga de procesamiento la ejecuta el servidor, aligerando así los tiempos en procesos tediosos; podremos llevar ejecuciones tediosas en el servidor en segundo plano siendo transparente para los usuarios.

Además entre el resto de sus características destaca que es un lenguaje muy flexible, ya que nos obliga a seguir un estándar de programación(no es necesario que se implemente una PDO o una programación secuencial), y es un lenguaje considerado de alto rendimiento.

Este lenguaje ha sido utilizado para dinamizar la plataforma web. Como principales funciones ha tenido la captación y reproducción de la información guardada en base de datos. En este caso se ha utilizado la versión 5.6.10.

7.1.1.2 Java

Java es un lenguaje de programación compilado dirigido a objetos. Desarrollado por James Gosling, que lo desarrollo para la compañía SunMicrosystems. Fue publicado en 1995, y más tarde fue adquirido por Oracle en 2010.

La principal característica de este lenguaje de programación es la portabilidad, pues una vez compilado puede ser ejecutado en cualquier equipo que tenga JVM (Java Virtual Machine) instalado.

Para este proyecto se ha utilizado para el sistema de recordación, ya que para ello se ha usado Mahout, que se basa en programación con Java. Además de que al ser una de las partes mas tediosas de configurar del proyecto, poder aprovechar la portabilidad era un gran punto a favor.

7.1.1.3 HTML

HTML (HyperText Markup Language) es un lenguaje de marcado para la construcción de páginas web. A partir de este lenguaje se puede dividir y dar forma a todos los elementos de una página web ya que es un lenguaje que hace uso de etiquetas para la distribución de elementos dentro de la página y de referencias para la introducción de elementos como imágenes, vídeos, etc.

El primer diseño del lenguaje de programación fue presentado en 1991, por parte de Berners-Lee. A partir de ahí se ha convertido en el lenguaje más popular para la construcción de páginas web.

Para este proyecto se ha utilizado para la parte de la plataforma web, pues todas los elementos visuales de la plataforma han sido desarrollados con este lenguaje de programación.

7.1.1.4 CSS

CSS (cascading style sheets) es un lenguaje usado para crear hojas de estilo, que se utilizan para dar estilo y forma las estructuras HTML. La objetivo principal de este lenguaje en su construcción era separar las hojas de estilos de los ficheros de la estructura de la página.

La primera versión de CSS fue propuesta por Håkon Wium Lie en 1994 y fue incluido en World Wide Web Consortium (W3C) en 1996.

Para este proyecto se ha utilizado para la parte de la plataforma web, dándole estilo y estructura a todos los elementos HTML mencionados anteriormente.

7.1.2 Frameworks, Bibliotecas y herramientas

7.1.2.1 MySQL

MySQL es una herramienta de gestión de base de datos relacionales. Rápida en las operaciones de lectura, con baja probabilidad de corrupción de la información y segura.

Una de las principales razones por la que se escogió esta base de datos es porque la conexión en PHP y en Java (dos de los lenguajes de programación utilizados) es sencilla, además al ser la plataforma un entorno intensivo en lectura de base de datos MySQL se convierte en un gestor de base de datos ideal para el proyecto.

7.1.2.2 Apache Mahout

Apache Mahout, es una librería de Machine learning, diseñada para ser escalable y robusta. Además de ser una herramienta altamente eficiente, pues funciona sobre Hadoop, una herramienta de Big Data, que se explicará un poco más adelante.

Apache Mahout fue elegida porque provee de algoritmos de clustering, clasificación y lo que es más importante en este proyecto, algoritmos de recomendación. Estos algoritmos ya desarrollados crean el vecindario de cada usuario a partir de los datos de las votaciones de todos los usuarios.

7.1.2.3 Apache Hadoop

Apache Hadoop es un framework que soporta aplicaciones distribuidas y fue creado principalmente como herramienta de Big Data, pudiendo procesar miles de nodos y petabytes de datos. La tecnología de Hadoop se basa en su sistema de archivos HDFS (Hadoop Distributed File System)[15], que es un sistema de archivos distribuido, escalable y portátil, que a groso modo se basa en la distribución de los datos en los diferentes nodos.

Hadoop fue creado por Doug Cutting, con la idea de apoyar la distribución del proyecto de motor de búsqueda Nutch.

En este proyecto es usado, por una parte para el sistema de recomendación pues Mahout se ejecuta sobre Hadoop y por otra parte, Solr (el motor de búsqueda de nuestra plataforma) estaba basado también en Hadoop. Es decir, en este proyecto no interviene de forma totalmente directa, pero es una pieza muy importante del mismo.

7.1.2.4 Lucene y Apache Solr

Apache Solr es un motor de búsqueda basado en Lucene, que incorpora APIs de JSON y XML y una interfaz de usuario.

Fue desarrollado por CNET Networks a finales de 2004 como proyecto interno, pero en el año 2006 CNET Networks decidió donar el código a la fundación Apache.

Por otra parte y como anteriormente comentábamos está basado en Lucene, que es una API de recuperación de información, capaz de indexar y buscar a texto completo en ficheros, dado que Lucene pone un motor de búsqueda de “crawling”[14]. De este modo Lucene es la herramienta con la cuál se recupera la información a buscar y Apache Solr nos provee de la manera más sencilla para recibir la información y poder procesarla.

7.1.2.5 Bootstrap

Es un framework para diseño de sitios y aplicaciones web. Es una herramienta para HTML, JavaScript y CSS que contiene plantillas de diseño, ya estructurados, con estilos, animaciones y funcionalidad. El objetivo del desarrollo de este framework no fue otro que el de agilizar la tarea de los desarrollados Front-end, ya que creaban interfaces de usuario con inconsistencias y una carga de trabajo alta en su mantenimiento.

La primera versión fue lanzada en Agosto de 2011 por parte de Mark Otto y Jacob Thornton.

En el proyecto ha realizado la función que sus desarrolladores tenían en mente en el momento de su creación, la de agilizar la tarea de creación del Front-end.

7.1.2.6 Maven

Maven es una herramienta para la gestión y construcción de proyectos Java. Permite gestionar e incluir bibliotecas de forma rápida y sencilla a través de un fichero XML[11].

Maven utiliza un Project Object Model (POM) para describir el proyecto de software a construir, sus dependencias de otros módulos y componentes externos, y el orden de construcción de los elementos. Viene con objetivos predefinidos para realizar ciertas tareas claramente definidas, como la compilación del código y su empaquetado.

En este proyecto ha sido utilizado para facilitar la ardua tarea de configuración del sistema de recomendación con Mahout.

7.2 *Fases de desarrollo*

En este capítulo podremos encontrar todas las fases de desarrollo que se han seguido y detalles del desarrollo que han ido apareciendo de modo en el que la implementación iba avanzado. Cabe destacar que para

7.2.1 Base de datos

El primer paso del desarrollo y teniendo en cuenta que iba a ser una plataforma dinámica fue crear la base de datos.

Como vimos en el apartado 4.1.1 se trata de una base de datos relacional[13] compuesta de siete tablas. Podemos encontrar 5 tablas con datos íntegros mientras que otras dos de ellas existen para satisfacer la relación n a n entre dos tablas. Por otra parte las tablas unidas por una relación 1 a n basan dicha relación en una primary key que siempre es de su identificador.

7.2.2 Parseo web de artículos

Con el fin de empezar teniendo un cantidad suficiente para poder empezar el proyecto se decidió parsear alguna web de artículos científicos en la que pudiésemos descargar los artículos y toda su información. Tras comprobar la estructura de varias web de investigación se seleccionó la que tenía una estructura más adaptada al parseo y se procedió a obtener la información con la ayuda de la biblioteca para PHP Simple HTML dom, que provee funciones para obtener el código fuente y todos los elementos que hay dentro del mismo a partir del tipo de etiqueta y la clase o el identificador del elemento. De este modo se procedió a la recuperación de dicha información y al posterior almacenado en base de datos.

7.2.3 Estructurar la plataforma web

Una vez tenemos estos datos ya podemos empezar a crear las vistas de nuestra plataforma. Antes de empezar a crearlas se estudio las secciones que tendría la web y la estructura e interfaz que tendría la misma. Al ser una plataforma sencilla y sin demasiadas secciones la mayoría de ellas fueron incluidas en el menú.

Para esta tarea se utilizó HTML y CSS. Con el fin de hacer la tarea más ágil se incorporó Bootstrap, lo que conllevó el uso de algunas de las plantillas de este framework para algunas de las secciones de la web. A parte de los ficheros CSS de Bootstrap también se creó un único fichero CSS en el que se incorporaron todos los estilos personalizados de la web.

Para facilitar la tarea y sabiendo que íbamos a usar PHP se crearon ficheros que contenían los enlaces con los CSS y Bootstrap para incorporarlos de la manera más sencilla posible a cada página de la plataforma.

7.2.4 Dinamización de la plataforma

Con el fin de hacer la página dinámica se empezaron a desarrollar algunas funciones específicas para este objetivo. Todas ellas se encuentran en un mismo fichero que contiene tanto funciones para insertar como para obtener datos.

Un punto importante sobre este tema es la creación de un fichero con la configuración para la conexión a base de datos, en la que podremos cambiar la información de forma sencilla en caso de ser necesario y además al tener un único archivo para esto es más sencillo de esconder en caso de que fuese necesario, pues contiene información muy sensible.

```
$dbhost = 'localhost';
$dbuser = 'talend';
$dbpass = 'talend';
$dbname = 'science_db';

function connect($dbhost, $dbuser, $dbpass, $dbname) {
    $connection = mysqli_connect($dbhost, $dbuser, $dbpass, $dbname);
    if ( mysqli_connect_errno() ) die( 'error: ' . mysqli_connect_error() );
    mysqli_query( $connection, "SET NAMES utf8" );

    return $connection;
}
```

Por otra parte podemos encontrar el control de sesiones también separado de las funciones. Para el control de las sesiones se ha utilizado la herramienta \$_SESSION de PHP para el almacenamiento de toda la información de la sesión del usuario.

```
session_start();|
$alert = false;
if ($_POST['init_session']) {
    $password = sha1($_POST['password']);
    $sql = "SELECT * FROM Usuario WHERE email='".$_POST['email']."' AND password = '".$_password.'";
    $result = mysqli_query($connection, $sql);
    if ( mysqli_num_rows( $result ) != 0 )
    {
        $result = mysqli_fetch_assoc($result);
    }
    if($result!=0) {
        // Iniciamos sesión
        $_SESSION['id'] = $result['id'];
        $_SESSION['email'] = $_POST['email'];
        $_SESSION['sess_ID'] = sha1($_POST['password']);
        $alert = "Acceso correcto";
        header('Location: http://localhost:8888/science/views/main.php');
    } else {
        // Si no son correctas devolvemos false
        $alert = "Las credenciales de acceso no son correctas";
    }
} else {
    $alert = "Algo ha ido mal...";
}
```

Como se ve en la captura de pantalla se guardan varios datos para asegurarnos de la autenticidad del usuario y no se pueda rellenar de forma ilícita el contenido de las variables de sesión. Además podemos ver como la contraseña se utiliza cifrada con SHA1, para que no puede ser roba.

Tras implementar todas las funciones y las sesiones se procedió a incorporar la información dinámica a cada página.

7.2.5 Motor de búsqueda

Una vez finalizadas todas las tareas anteriormente citadas se procedió a la construcción, configuración e implementación del motor de búsqueda.

Tras realizar un estudio los principales motores de búsqueda disponibles, se llegó a la conclusión de la herramienta más completa era Solr, por su gran rendimiento, además de ser sencilla de configurar puesto que cuando te descargas los ficheros tan solo hay que lanzar un comando ya estará en funcionamiento. Una vez arrancado el servicio dispone de una interfaz web en la que se pueden configurar y crear colecciones. Estas colecciones se utilizan para almacenar la información sobre los ficheros indexados.

Dentro la interfaz web se generan las colecciones, señalando los nodos que quieres que tenga, las réplicas en cada nodo, el nombre y la configuración, en cuanto a campos almacenados, que quieres que tenga. En nuestro caso elegiremos un modelo de datos sencillo puesto que lo que nos interesa es que al indexar los ficheros son parseados, a través de la librería que incorpora Solr llamada Tika[12], y luego podremos hacer consultas sobre el contenido de los mismos, así que solo guardaremos el path donde son guardados.

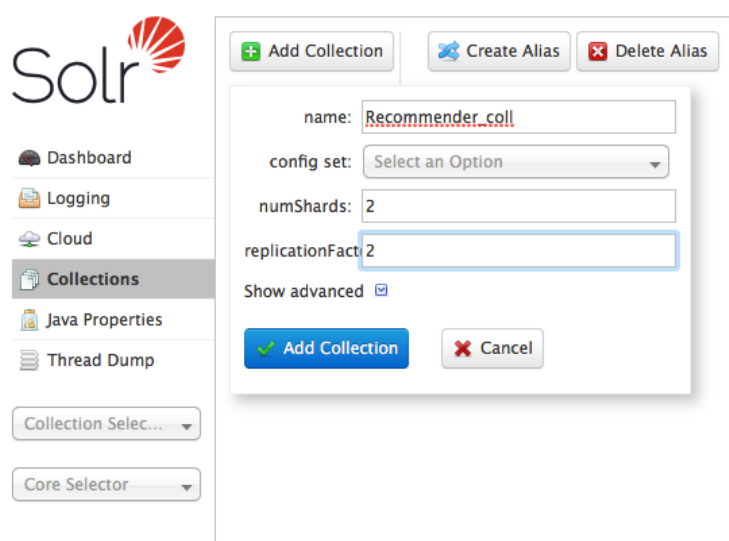


Imagen 30 - Creación de colección en Solr

Una vez creado la colección se procedió a indexar los archivos que ya teníamos del parseo de la web de artículos científicos a través del terminal. Una vez indexados todos estos ficheros estamos listo para hacer búsquedas sobre nuestros ficheros a través de una petición a través de peticiones HTTP que nos devuelven una respuesta en JSON con todas las informaciones. De esta manera es como es introducido el motor de búsqueda en el buscador de la plataforma web, el cuál realiza una petición HTTP a través de un formulario y parsea y muestra la información que recibe de la respuesta, tal y como se muestra en el siguiente fragmento del código:

```
if ($_POST) {
    if ($_POST['busca']) {
        // Tratamos la información para que tenga el formato adecuado...
        $data = str_replace(' ', "%20", $_POST['busca']);

        // Preparamos la consulta...
        $request = "http://192.168.1.102:8983/solr/Recommender/select?indent=on&q=".$data."&wt=json";

        // Configuramos los parametros de curl correctamente para la consulta...
        $ch = curl_init();
        curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
        curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
        curl_setopt($ch, CURLOPT_URL, $request);
        $result = curl_exec($ch);
        curl_close($ch);

        // Decodificamos el JSON de respuesta...
        $obj = json_decode($result, true);
        $result = array();
        // Obtenemos todos los resultados y los almacenamos en un array para imprimirlos en la sección
        // en la que deben de ir los resultados...
        for ($i = 0; $i < $obj['response']['numFound']; ++$i) {
            $art = getArticleByPath($connection, substr($obj['response']['docs'][$i]['id'], 48));
            array_push($result, $art);
        }
    }
}
```

7.2.6 Sistema de recomendación

Para evitar el *inicio frío*, se decidió hacer un sistema de recomendación mixto, es decir, un sistema de recomendación que tuviese tanto recomendaciones basadas en filtrado colaborativo (a partir de las votaciones de otros que tiene votos parecidos a los tuyos) y un filtrado basado en contenido o ítem (a partir de etiquetas que comparte el usuario y el artículo).

Para el filtrado basado en contenido se procede de la siguiente forma:

- 1) Cuando un usuario se da de alta en la plataforma se le pide que escriba con una o dos palabras el campo de su trabajo, su tema preferido y el tema de su última investigación. De este modo guardamos en base de datos información sobre los intereses del usuario.

- 2) Utilizando nuestro motor de búsqueda realizamos una búsqueda con la información que nos ha sugerido el usuario cuando se ha dado de alta (también llamado etiquetas).
- 3) Se le devuelven al usuario las recomendaciones a partir de esa búsqueda.

Por otra parte encontramos el sistema de recomendación basado en filtrado colaborativo, que se explica mas detenidamente a continuación.

Como se comentó anteriormente, para esta funcionalidad del proyecto se eligió Apache Mahout, el cuál implica cierta problemática a la hora de configurarlo correctamente, pues que requiere algunas librerías que son en ocasiones algo conflictivas a la hora de integrarlas correctamente, así que se optó por buscar una herramienta que ayudase a la integración y compilación del mismo y se decidió que la mejor herramienta sería Maven. De este modo se incorporó Maven a los ficheros y se configuró el fichero 'pom.xml', donde se configurar las librerías que se van a utilizar, del siguiente modo:

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.technobium</groupId>
  <artifactId>recommender</artifactId>
  <packaging>jar</packaging>
  <version>1.0-SNAPSHOT</version>
  <name>recommender</name>
  <url>http://maven.apache.org</url>
  <dependencies>
    <dependency>
      <groupId>mysql</groupId>
      <artifactId>mysql-connector-java</artifactId>
      <version>5.1.6</version>
    </dependency>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>
    <dependency>
      <groupId>org.apache.mahout</groupId>
      <artifactId>mahout-core</artifactId>
      <version>0.9</version>
    </dependency>
    <dependency>
      <groupId>org.slf4j</groupId>
      <artifactId>slf4j-simple</artifactId>
      <version>1.7.7</version>
    </dependency>
  </dependencies>
</project>
```

De este modo y tras configurar algunos parámetro de Mahout se procedió a estructurar y desarrollar el sistema de recomendación. Para evitar sobrecargar el servidor con peticiones recurrentes de recomendaciones, teniendo en cuenta la carga que esto supone, se decide llevar a cabo una estrategia de generado de recomendaciones en

segundo plano guardándolas en base de datos. A continuación se explica con detalles el proceso:

- 1) Exportamos todas las votaciones a un fichero csv, ya que Mahout mejora su eficiencia al leer de un fichero antes que de una base de datos. Esta exportación se realizara con PHP como se muestra en el siguiente fragmento de código:

```
$sql = "SELECT id_usuario, id_articulo, puntuacion FROM Votacion";
$result = mysqli_query($connection, $sql);
$myfile = fopen("valorations.csv", "w") or die("Unable to open file!");
foreach ($result as $res) {
    $txt = $res['id_usuario'] . "," . $res['id_articulo'] . "," . $res['puntuacion'] . "\n";
    fwrite($myfile, $txt);
}
fclose($myfile);
```

- 2) Una vez realizado este volcado de las valoraciones en base de datos se procede a analizar esos datos con las funcionalidades que nos aporta Mahout de la siguiente manera:

```
// Borramos lo que había en las recomendaciones...
ResultSet rs = stmt.executeQuery("DELETE FROM Votacion");

tmt = conn.createStatement();
stmt2 = conn.createStatement();

Logger log = LoggerFactory.getLogger(BasicRecommender.class);

// Abrimos el fichero con las votaciones...
DataModel model = new FileDataModel(new File("input/valorations.csv"));

// Comprobamos la similitud entre usuarios...
UserSimilarity similarity = new EuclideanDistanceSimilarity(model);

// Creamos el vecindario del usuario con usuarios con los que comparte valoraciones...
UserNeighborhood neighborhood = new ThresholdUserNeighborhood(0.1,
    similarity, model);

// Se crea la recomendación...
UserBasedRecommender recommender = new GenericUserBasedRecommender(
    model, neighborhood, similarity);

// Insertamos todos las recomendaciones para cada usuario...
ResultSet rs = stmt.executeQuery("SELECT * FROM Usuario");
while ( rs.next() ) {
    id_usuario = rs.getString("id");
    List<RecommendedItem> recommendations = recommender.recommend(Integer.parseInt(id_usuario), 5);
    for (RecommendedItem recommendation : recommendations) {
        String sql = "INSERT INTO Recomendacion VALUES (" + id_usuario + "," + recommendation.getItemID() + ")";
        stmt2.executeUpdate(sql);
        log.info("User " + id_usuario + " might like the book with ID: "
            + recommendation.getItemID() + " (predicted preference : "
            + recommendation.getValue() + ")");
    }
}
```

- 3) Tras tener esto listo creamos el script donde ejecutaremos el fichero PHP, moveremos los ficheros a la carpeta donde lo obtiene el fichero Mahout y además ejecutaremos el ejecutable de Java con el sistema de recomendación.
- 4) Creamos un cron para ejecutar a las 3.00 de la mañana el script que hemos mencionado anteriormente.

Un detalle a tener en cuenta sobre las recomendaciones basadas en filtrado colaborativo es que el usuario recibe veinte recomendaciones basadas en filtrado colaborativo en el mejor de los casos, es decir, teniendo la información suficiente para tener tal número de recomendaciones. Estas recomendaciones se actualizan a diario pudiendo desaparecer si ha valorado estos artículos o bien si aparecen artículos que son considerados de mayor interés por el usuario.

7.3. Pruebas

Las pruebas se realizaban tras la finalización de cada objetivo, llevando a cabo pruebas de uso, en las que se pulían bugs y problemas derivados de la implementación, asegurándonos así del correcto funcionamiento. Tras la finalización de la implementación se realizaron pruebas en toda la plataforma, es decir, se simuló el comportamiento normal de un usuario y se encontraron dos problemas o *bugs* principalmente:

- El listado de los usuarios mejor valorados listaba los usuarios con mejores notas, pero tan solo lo ordenaba únicamente por la media, así que se procedió a cambiar la *query SQL* para que además de por la media también ordenase los que más votaciones habían recibido.
- Al valorar un artículo y volver al buscador debíamos retroceder en dos ocasiones pues al insertar la valoración la pantalla se refrescaba al procesar el formulario de votación. Se procedió a insertar las votaciones con AJAX, de modo que la página no se refrescase y volviese al buscador de forma directa.

Tras la corrección de estos errores se dio por verificado el correcto funcionamiento de la plataforma cumpliendo todos los requisitos funcionales de manera correcta.

8. Resultados

8.1 La plataforma

En este apartado veremos cómo ha quedado la plataforma, con algunos pantallazos sobre la web y algunas muestras de información.

8.1.1 Navegabilidad

Como ya hablamos anteriormente, se ha buscado que la navegabilidad sea lo más fluida posible dividiendo todas las secciones lo máximo posible e integrándolas en el menú superior y en un pequeño sub-menú lateral , como se puede observar a continuación:



Imagen 31 - Menú

8.1.2 Secciones principales

8.1.2.1 Buscador

Se ha optado por un diseño sencillo y sencillo de ver, como se muestra a continuación. Aquí es donde podremos hacer la búsqueda sobre artículos que nos interesen.

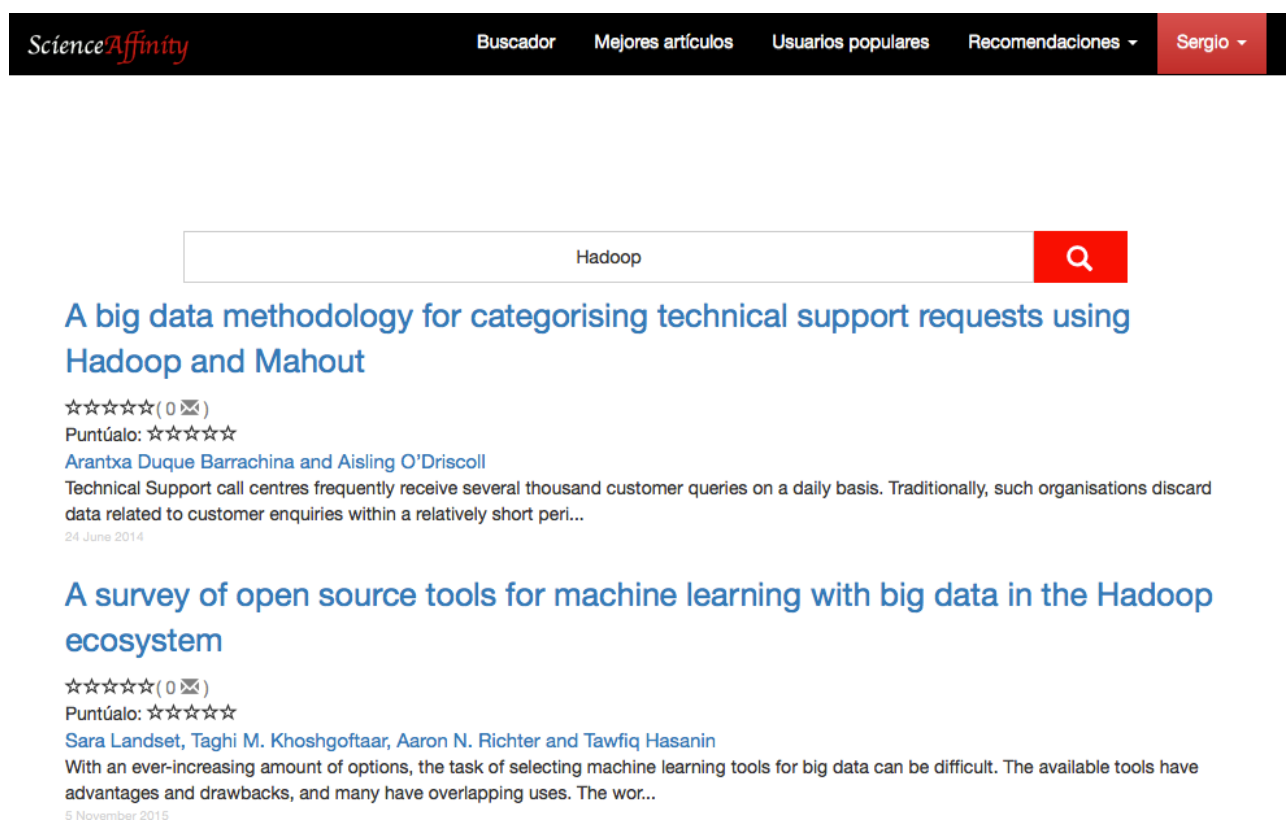


Imagen 32 - Buscador

8.1.2.1 Recomendaciones

Como se explicó en secciones anteriores las recomendaciones están divididas en secciones:

- 1) Recomendaciones basadas en contenido:



Recomendaciones basadas en tus intereses

* Estas recomendaciones se basan en los intereses que nos dijiste cuando te diste de alta. Si crees que se aproxima poco a tus intereses siempre puedes ir a tu perfil y cambiarlos!.

Big data analytics: a survey

☆☆☆☆☆ (0 )

Puntúalo: ☆☆☆☆☆

[Chun-Wei Tsai](#), [Chin-Feng Lai](#), [Han-Chieh Chao](#) and [Athanasios V. Vasilakos](#)

The age of big data is now coming. But the traditional data analytics may not be able to handle such large quantities of data. The question that arises now is, how to develop a high performance *platform* to effic...

1 October 2015

Reaping the benefits of big data in telecom

☆☆☆☆☆ (0 )

Puntúalo: ☆☆☆☆☆

[Jacques Bughin](#)

We collect big data use cases for a representative sample of telecom companies worldwide and observe a wide and skewed distribution of big data returns, with a few companies reporting large impact for a long t...

28 July 2016

Imagen 33 - Recomendaciones basadas en contenido

2) Recomendaciones basadas en filtrado colaborativo:

Recomendaciones para tí

* Recuerda que estás son tus 10 recomendaciones para hoy, es posible que si no las puntuas podrán cambiar mañana, en caso de que las puntúes ya no volverán a aparecer en esta lista pero podrás consultarlos en tu lista de artículos votados.

[Mining Chinese social media UGC: a big-data framework for analyzing Douban movie reviews](#)

★★★★★ (3 ✉)

Tu puntuación: ★★★★★

[Jie Yang and Brian Yecies](#)

Analysis of online user-generated content is receiving attention for its wide applications from both academic researchers and industry stakeholders. In this pilot study, we address common Big Data problems of...

13 January 2016

[A novel algorithm for fast and scalable subspace clustering of high-dimensional data](#)

★★★★★ (2 ✉)

Puntúalo: ☆☆☆☆☆

[Amardeep Kaur and Amitava Datta](#)

Rapid growth of high dimensional datasets in recent years has created an emergent need to extract the knowledge underlying them. Clustering is the process of automatically finding groups of similar data points...

12 August 2015

Imagen 34 - Recomendaciones por filtrado colaborativo

8.1.3 Secciones secundarias con puntos de interés

8.1.3.1 Alta usuario

Esta sección es secundaria pero tiene una parte muy importante en el sistema de recomendación, las etiquetas que se le asignan al usuario cuando se da de alta, que se piden del siguiente modo:

Tus datos de acceso

Email

Contraseña

Tus intereses

En esta sección tan solo tienes que introducir palabras relacionadas al tema sobre el que trabajas y sobre lo que esperas encontrar información

Trabajo

Interés

Última investigación

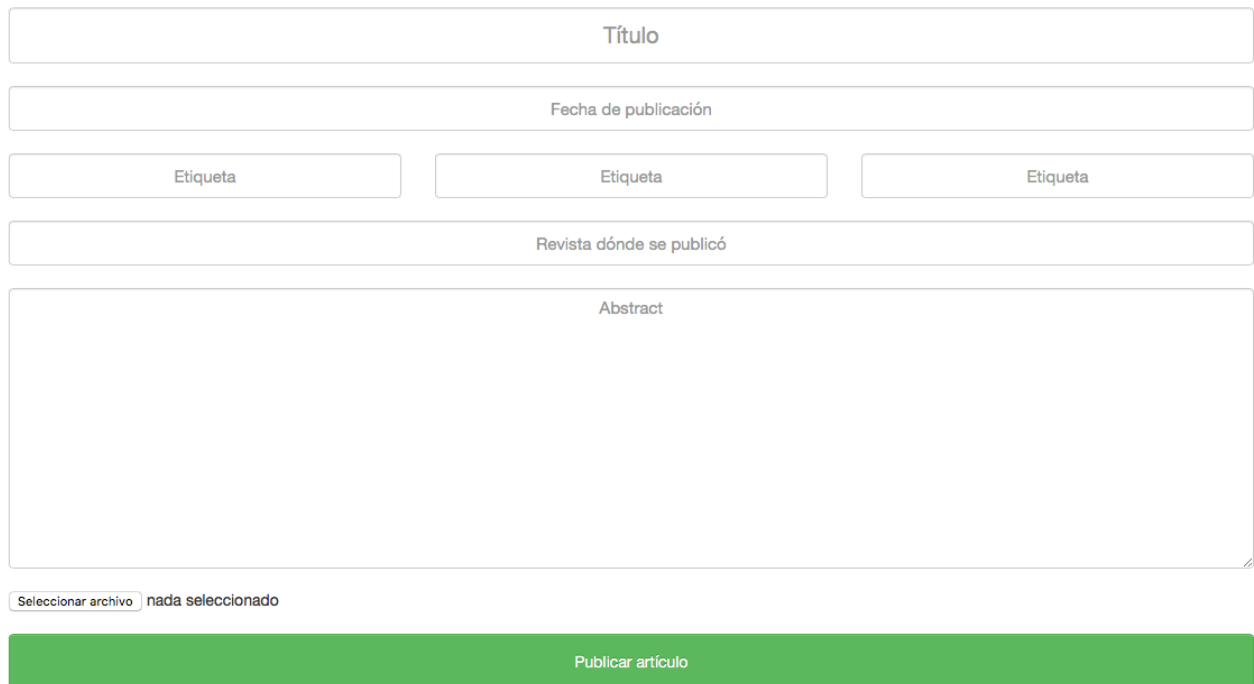
Esto ha sido todo!

¡Dadme de alta!

Imagen 35 - Registro de usuarios

8.1.3.2 Publicar nuevo artículo

Esta sección también es importante pues es la manera de nutrir de información nuestro sistema de recomendación. Se ha desarrollado pensando en la manera más rápida y ágil de publicar el artículo y pidiendo la información básica.



Formulario para publicar un nuevo artículo. El formulario contiene los siguientes campos:

- Título
- Fecha de publicación
- Etiqueta (tres campos)
- Revista dónde se publicó
- Abstract

Debajo de los campos, hay un botón "Seleccionar archivo" y el texto "nada seleccionado".

En la parte inferior, hay un botón verde "Publicar artículo".

Imagen 36 - Publicar nuevos artículos

9. Conclusiones

9.1 Objetivos alcanzados

En este apartado retomaremos los objetivos planteados al principio de la memoria para analizar si se han cumplido.

Objetivo 1	Dar de alta usuarios
Tipo	Obligatorio
Conseguido	Sí
Detalles	La plataforma permite darse de alta a nuevos usuarios y gestiona las sesiones de los mismos.

Objetivo 2	Introducción de artículos en la plataforma
Tipo	Obligatorio
Conseguido	Sí
Detalles	La plataforma permite subir y publicar nuevos artículos.

Objetivo 3	Búsqueda sobre temas
Tipo	Obligatorio
Conseguido	Sí
Detalles	La plataforma permite realizar búsquedas a través del buscador y mostrar los resultados a través del motor de búsqueda.

Objetivo 4	Permitir votaciones a artículos
Tipo	Obligatorio
Conseguido	Sí
Detalles	El usuario puede realizar votaciones cuando se listan y se consulta la información detallada de los artículos.

Objetivo 5	Realizar recomendaciones
Tipo	Obligatorio
Conseguido	Sí
Detalles	El sistema genera recomendaciones, tanto por contenido como por filtrado colaborativo.

Objetivo 6	Listar los artículos con mejores valoraciones
Tipo	Opcional
Conseguido	Sí
Detalles	La plataforma tiene una sección de 'mejor valorados' en el que podemos encontrar los artículos que han recibido mejores votaciones.

Objetivo 7	Nutrirse de otras fuentes al comienzo
Tipo	Opcional
Conseguido	Sí
Detalles	Al principio del proyecto se parseó otra plataforma de artículos de donde se extrajeron ficheros y artículos.

Objetivo 8	Ser eficiente
Tipo	Opcional
DescripciónConseguido	Sí
Detalles	Se han implementado los procesos con más carga de forma que afecten lo menos posible a la carga de trabajo del servidor pero realmente no se puede decir con total seguridad que la plataforma no tendrá picos de trabajo que puedan afectar a la experiencia del usuario, ya que aún no ha sufrido pico de estrés.

9.2 Futuras vías

Como todo proyecto siempre se pueden encontrar numerosas mejoras que podrían proveer una mejor experiencia al usuario y hacer el proyecto más útil, desde mejoras de eficiencia hasta mejoras funcionales. Los siguientes pasos que podría tener nuestro proyecto serían:

- 1) Añadir relaciones entre usuarios: una posible mejora que haría más interactiva la plataforma, será añadir parcialmente un red social, en la que los usuarios pudiesen agregarse como ‘amigos’ y seguir los progresos de cada una de las personas que están en su red de amigos.
- 2) Mejorar el seguimiento de todos tus artículos que son publicados por el usuario con notificaciones, ya sea vía email o simplemente añadiendo un centro de notificaciones en la plataforma.
- 3) Realizar las recomendaciones por filtrado colaborativo en tiempo real.
- 4) Corrección de bugs que aparecerán con el uso
- 5) Mejoras en seguridad

10. Bibliografía

10.1 Referencias

- [1] - Sergio Cleger Tamayo (2012). Diseño y validación de modelos para sistemas de recomendación. Universidad de Granada. Granada.
- [2] - Luis M. de Campos, Juan M. Fernández-Luna, Juan F. Huete, Miguel A. Rueda-Morales. Uso de conocimiento estructurado en un sistema de recomendación basado en contenido. Departamento de Ciencias de la Computación e Inteligencia Artificial. E.T.S.I. Informática, Universidad de Granada. Granada.
- [3] - Almudena Ruiz Iniesta - Sistemas de recomendación. Presente y futuro de la web. Universidad Complutense de Madrid. Madrid.
- [4] - JA Konstan (2012)- Recommender systems: from algorithms to user experience - Springer Science+Business Media.
- [5] - Diana C. Ramírez Martínez, Luis C. Martínez Ruiz, Oscar F. Castellanos Dominguez. Divulgación y difusión del conocimiento. Universidad Nacional de Colombia. Colombia.
- [6] - Constantine, L. L., Lockwood, L. A. D. (1999). Software for Use: A Practical Guide to the Models and Methods of Usage - Centred Design. Addison – Wesley.
- [7] - Miñones Crespo, R. El exceso o sobrecarga de información en la sociedad de la información. Universidad de La Coruña.
- [8] - Anónimo. Programación funcional. En Wikipedia. Recuperado el 6 de Septiembre de 2016 de [https://es.wikipedia.org/wiki/Programación funcional](https://es.wikipedia.org/wiki/Programación_funcional)
- [9] - Anónimo. Programación orientada a objetos. En Wikipedia. Recuperado el 6 de Septiembre de 2016 de [https://es.wikipedia.org/wiki/Programación orientada a objetos](https://es.wikipedia.org/wiki/Programación_orientada_a_objetos).
- [10] - Anónimo. PHP. En wikipedia. Recuperado el 8 de Septiembre de 2016. <https://es.wikipedia.org/wiki/PHP>
- [11] - Anónimo - What is Maven?. <https://maven.apache.org/what-is-maven.html>. Accedido el 8 de Septiembre de 2016.
- [12] - Targett C. Uploading Data with Solr Cell using Apache Tika. <https://cwiki.apache.org/confluence/display/solr/Uploading+Data+with+Solr+Cell+using+Apache+Tika>-. Accedido el 8 de Septiembre de 2016.
- [13] - Anónimo. Base de datos relacional. Recuperado el 8 de Septiembre de 2106. [https://es.wikipedia.org/wiki/Base de datos relacional](https://es.wikipedia.org/wiki/Base_de_datos_relacional)

[14] - Anónimo. Araña web. Recuperado el 8 de Septiembre de 2106. https://es.wikipedia.org/wiki/Araña_web

[15] - Borthakur D. HDFS Architecture Guide. Recuperado el 8 de Septiembre de 2016. https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html

10.2 Enlaces de interés

- Web de Mahout:

<http://mahout.apache.org/>

- Web de Hadoop:

<http://hadoop.apache.org/>

- Web de Lucene:

<http://lucene.apache.org/>

- Web de Solr:

<http://lucene.apache.org/solr/>

- Web de PHP:

<http://php.net/>

- Documentación sobre CSS

<http://w3schools.com/css/>

- Documentación sobre HTML

<http://w3schools.com/html/>

- Web de Bootstrap

<http://getbootstrap.com/>

11. Anexos

11.1 Glosario de términos

11.1.1 Términos

A continuación se presentan los términos específicos del dominio del problema con una definición.

- **Machine learning:** Hace referencia al aprendizaje automático de los computadores, es decir, al conjunto de algoritmos que proveen al computador de cierto tipo de aprendizaje y que le permiten reproducir algún tipo de comportamiento a través de la información que se le da a esos algoritmos.
- **Crawling:** Es un algoritmo de búsqueda que inspecciona todos los elementos que entra en el rango de la búsqueda y que busca dentro de ellos información que puede enlazarle con más contenido relacionado.
- **Interfaz de usuario (UI):** Es el medio con que el usuario puede comunicarse con nuestro programa.
- **Parsear:** En argot informático se refiere al análisis gramatical de fichero.
- **World Wide Web Consortium:** es un consorcio internacional que genera recomendaciones y estándares que aseguran el crecimiento de la World Wide Web a largo plazo.
- **Mockup:** Maqueta del diseño que tendrá una página web o cualquier documento que necesito una estructura y diseño.
- **Bug:** Error.
- **Query:** Consulta.

11.1.2 Acrónimos

A continuación se listan los acrónimos o abreviaturas utilizados durante esta memoria escrita:

- **CSS:** *cascading style sheets*, hoja de estilos. Lenguaje de programación.
- **HTML:** *HyperText Markup Language*, Lenguaje de marcas de hipertexto. Lenguaje de programación.
- **PDF:** Portable Document Format, formato de documentación portable. Formato de documentos.
- **PHP:** Hypertext Preprocessor, Procesador de hipertexto. Lenguaje de programación.
- **DB:** *Database*, Base de datos.
- **UI:** User Interface, Interfaz de usuario.