# UNIVERSIDAD DE OVIEDO

ESCUELA DE INGENIERÍA INFORMÁTICA

## TRABAJO FIN DE GRADO

"Diseño e implementación de una aplicación web para clasificación de documentos mediante técnicas de aprendizaje automático"

| | |
|---|---|
| | **DIRECTOR: Juan Luis Mateo Cerdán** |
| **Vº Bº del Director del Proyecto** | **AUTOR: Carlos Sanabria Miranda** |

# Agradecimientos

En primer lugar, quiero dar las gracias a mi familia por su continuo apoyo en este largo camino. En especial, a mis padres, mi hermano y mis abuelos, sin los cuáles nada de esto hubiera sido posible. Siempre me habéis guiado por el buen camino, así que el mérito de todo esto también es vuestro.

Agradecer también a mi novia Cristina por aguantarme durante estos últimos años, los cuales han sido complicados, pero ha merecido la pena. En muchas ocasiones me has dado la motivación que me faltaba para seguir adelante.

También me gustaría dar las gracias a mi tutor, Juan Luis, por su enorme ayuda y dedicación. Desde el primer momento me has aconsejado y guiado para llevar este proyecto a buen puerto, lo cual considero que hemos conseguido con creces. Si se han logrado los objetivos planteados inicialmente es, en gran parte, gracias a ti.

De igual forma, quiero agradecer a Accenture por darme la posibilidad de realizar parte de este Trabajo Fin de Grado durante el transcurso de las prácticas de empresa. En específico, quiero mencionar a Violeta, quién ha sido mi tutora durante las prácticas, y ha aportado la idea base del proyecto. Gracias por ayudarme a sacar adelante la parte de generación de los modelos.

Por último, dar las gracias a mis profesores y compañeros (muchos de los cuales considero amigos) durante estos 4 años de carrera, tanto los que he tenido en mis 2 primeros años en la Escuela Politécnica de Ingeniería de Gijón, como los que he tenido en estos 2 últimos años en la Escuela de Ingeniería Informática de Oviedo. Me habéis hecho el camino más fácil, y he aprendido mucho de todos vosotros. Me quedo con todos los momentos que hemos compartido juntos.

# Resumen

El presente Trabajo Fin de Grado tiene como objetivo principal el desarrollo de una herramienta que permita identificar las temáticas presentes en una colección de documentos de texto, para luego poder clasificar dichos documentos dentro de las temáticas identificadas, así como resumir su contenido. A la llegada de un nuevo documento de texto, la herramienta lo clasificará dentro de una de las temáticas identificadas previamente, y realizará un resumen extractivo del mismo.

Además del diseño y la implementación de esta herramienta, el otro gran objetivo de este proyecto ha sido desarrollar una aplicación web que permita interactuar con ella a través de una interfaz gráfica, y que está dividida en dos subsistemas: un backend que sirve una API REST con la funcionalidad de la herramienta, implementado en Flask, y un frontend que consume la API REST, implementado en Angular.

# *Palabras Clave*

Clasificación de texto, Aprendizaje automático, Aprendizaje no supervisado, Procesamiento de lenguaje natural, Extracción de topics, Generación de resúmenes de textos, Angular, Flask.

# Abstract

The present Final Degree Project has as main object the development of a tool that allows to identify the topics present in a collection of text documents, to later be able to classify those documents into the identified topics, as well as summarize its content. Upon arrival of a new document, the tool will classify it inside one of the previously identified topics, and it will generate an extractive summary of the document.

In addition to the design and implementation of this tool, the other main objective of this project has been developing a web application that allows to interact with it through a graphical user interface, that it's divided in two subsystems: a backend that serves a REST API with the tool functionality, implemented in Flask, and a frontend that consumes the REST API, implemented in Angular.

# Keywords

Text classification, machine learning, unsupervised learning, Natural Language Processing, Topic's extraction, Text summarization, Angular, Flask.

# Índice General

**CAPÍTULO 1. MEMORIA DEL PROYECTO ..... 13**

1.1 RESUMEN DE LA MOTIVACIÓN, OBJETIVOS Y ALCANCE DEL PROYECTO ..... 13
1.2 RESUMEN DE TODOS LOS ASPECTOS ..... 14

**CAPÍTULO 2. INTRODUCCIÓN ..... 15**

2.1 JUSTIFICACIÓN DEL PROYECTO ..... 15
2.2 OBJETIVOS DEL PROYECTO ..... 16
2.3 ESTUDIO DE LA SITUACIÓN ACTUAL ..... 17
*2.3.1 Sistemas similares ..... 17*
*2.3.2 Evaluación de Alternativas ..... 18*

**CAPÍTULO 3. ASPECTOS TEÓRICOS ..... 21**

3.1 NLP ..... 21
3.2 N-GRAM ..... 21
3.3 WORD EMBEDDING ..... 22
3.4 LDA ..... 22
3.5 TEXTRANK ..... 23
3.6 GENSIM ..... 24
3.7 MALLET ..... 24
3.8 FLASK ..... 24
3.9 REST ..... 25
3.10 ANGULAR ..... 25
3.11 SPA ..... 25

**CAPÍTULO 4. PLANIFICACIÓN DEL PROYECTO Y PRESUPUESTO INICIALES ..... 26**

4.1 PLANIFICACIÓN INICIAL ..... 26
*4.1.1 Estudios iniciales ..... 28*
*4.1.2 Documentación inicial ..... 28*
*4.1.3 Planificación y presupuestos iniciales ..... 28*
*4.1.4 Análisis ..... 29*
*4.1.5 Diseño ..... 29*
*4.1.6 Implementación ..... 29*
*4.1.7 Implementación de las pruebas ..... 30*
*4.1.8 Despliegue ..... 30*
*4.1.9 Resto de la documentación ..... 30*
4.2 PRESUPUESTO DE COSTES INICIAL (RESUMIDO) ..... 31

**CAPÍTULO 5. ANÁLISIS ..... 32**

5.1 DEFINICIÓN DEL SISTEMA ..... 32
*5.1.1 Determinación del Alcance del Sistema ..... 32*
5.2 REQUISITOS DEL SISTEMA ..... 33
*5.2.1 Obtención de los Requisitos del Sistema ..... 33*
*5.2.2 Identificación de Actores del Sistema ..... 37*
*5.2.3 Especificación de Casos de Uso ..... 37*
5.3 IDENTIFICACIÓN DE LOS SUBSISTEMAS EN LA FASE DE ANÁLISIS ..... 39

*5.3.1 Descripción de los Subsistemas* ... *39*
*5.3.2 Descripción de los Interfaces entre Subsistemas* ... *39*
5.4 DIAGRAMA DE CLASES PRELIMINAR DEL ANÁLISIS ... 40
*5.4.1 Diagrama de Clases* ... *40*
*5.4.2 Descripción de las Clases* ... *40*
5.5 ANÁLISIS DE INTERFACES DE USUARIO ... 46
*5.5.1 Descripción de la Interfaz* ... *46*
*5.5.2 Descripción del Comportamiento de la Interfaz* ... *49*
*5.5.3 Diagrama de Navegabilidad* ... *49*
5.6 ESPECIFICACIÓN DEL PLAN DE PRUEBAS ... 50

**CAPÍTULO 6. DISEÑO DEL SISTEMA ... 51**

6.1 ARQUITECTURA DEL SISTEMA ... 51
*6.1.1 Diagramas de Componentes* ... *52*
*6.1.2 Diagramas de Despliegue* ... *53*
6.2 DISEÑO DE CLASES ... 55
*6.2.1 Diagrama de Clases* ... *55*
6.3 DISEÑO DE LA INTERFAZ ... 58
6.4 ESPECIFICACIÓN TÉCNICA DEL PLAN DE PRUEBAS ... 63
*6.4.1 Pruebas Unitarias automatizadas* ... *63*
*6.4.2 Pruebas de Usabilidad* ... *64*
*6.4.3 Pruebas de Adaptabilidad* ... *67*

**CAPÍTULO 7. IMPLEMENTACIÓN DEL SISTEMA ... 68**

7.1 ESTÁNDARES Y NORMAS SEGUIDOS ... 68
7.2 LENGUAJES DE PROGRAMACIÓN ... 69
*7.2.1 Python 3* ... *69*
*7.2.2 TypeScript* ... *69*
*7.2.3 HTML* ... *70*
*7.2.4 CSS* ... *70*
7.3 HERRAMIENTAS Y PROGRAMAS USADOS PARA EL DESARROLLO ... 71
*7.3.1 PyCharm* ... *71*
*7.3.2 WebStorm* ... *71*
*7.3.3 Postman* ... *71*
*7.3.4 Git* ... *71*
*7.3.5 Docker* ... *72*
*7.3.6 Vagrant* ... *72*
*7.3.7 Chrome* ... *72*
7.4 CREACIÓN DEL SISTEMA ... 73
*7.4.1 Problemas Encontrados* ... *73*
*7.4.2 Descripción Detallada de las Clases* ... *76*

**CAPÍTULO 8. DESARROLLO DE LAS PRUEBAS ... 77**

8.1 PRUEBAS UNITARIAS AUTOMATIZADAS ... 77
*8.1.1 Subsistema Topics and Summary* ... *77*
*8.1.2 Subsistema Web backend* ... *79*
8.2 ANÁLISIS ESTÁTICO DE CÓDIGO ... 81
*8.2.1 Subsistema Topics and Summary* ... *81*
*8.2.2 Subsistema Web backend* ... *82*
8.3 PRUEBAS DE USABILIDAD ... 83

8.4 PRUEBAS DE ADAPTABILIDAD ..... 91
*8.4.1 Resolución 1440 x 768 ..... 91*
*8.4.2 Resolución 1024 x 768 ..... 93*
*8.4.3 Resolución 700 x 768 ..... 96*
*8.4.4 Resolución 320 x 768 ..... 99*

**CAPÍTULO 9. MANUALES DEL SISTEMA ..... 103**

9.1 MANUAL DE INSTALACIÓN ..... 103
9.2 MANUAL DE EJECUCIÓN Y DESPLIEGUE ..... 104
9.3 MANUAL DEL PROGRAMADOR ..... 105
9.4 DOCUMENTACIÓN ADICIONAL ..... 106

**CAPÍTULO 10. CONCLUSIONES Y AMPLIACIONES ..... 107**

10.1 CONCLUSIONES ..... 107
10.2 AMPLIACIONES ..... 108
*10.2.1 Implementar las funcionalidades del administrador ..... 108*
*10.2.2 Probar los modelos de extracción de topics con más datasets ..... 108*
*10.2.3 Evaluar los resultados del algoritmo TextRank ..... 108*
*10.2.4 Probar otros algoritmos para la generación de los resúmenes ..... 109*
*10.2.5 Aumentar el número de mensajes de ayuda en la interfaz gráfica ..... 109*
*10.2.6 Generar los resúmenes de los documentos del dataset en batch ..... 109*
*10.2.7 Ampliar el subsistema Topics and Summary para que permita trabajar con datasets en español 109*
*10.2.8 Añadir internacionalización al frontend ..... 110*
*10.2.9 Hacer la interfaz web accesible ..... 110*
*10.2.10 Otras ampliaciones ..... 110*

**CAPÍTULO 11. PLANIFICACIÓN DEL PROYECTO Y PRESUPUESTO FINALES ..... 111**

11.1 PLANIFICACIÓN FINAL ..... 111
*11.1.1 Estudios iniciales ..... 112*
*11.1.2 Documentación inicial ..... 113*
*11.1.3 Planificación y presupuestos iniciales ..... 113*
*11.1.4 Análisis ..... 113*
*11.1.5 Diseño ..... 114*
*11.1.6 Implementación ..... 114*
*11.1.7 Implementación de las pruebas ..... 115*
*11.1.8 Resto de la documentación ..... 115*
11.2 PRESUPUESTO DE COSTES FINAL (RESUMIDO) ..... 116

**CAPÍTULO 12. REFERENCIAS BIBLIOGRÁFICAS ..... 117**

**CAPÍTULO 13. APÉNDICES ..... 119**

13.1 GLOSARIO Y DICCIONARIO DE DATOS ..... 119
13.2 CONTENIDO ENTREGADO EN EL ARCHIVO ADJUNTO ..... 122
13.3 CÓDIGO FUENTE ..... 123
13.4 PRESUPUESTO DE COSTES INICIAL (DETALLADO) ..... 124
*13.4.1 Modelo de empresa ..... 124*
*13.4.2 Explicación del presupuesto de costes inicial ..... 129*
13.5 PRESUPUESTO DE COSTES FINAL (DETALLADO) ..... 143

# Índice de Figuras

Figura 4.1: Comparativa de duración de las etapas del proyecto en la planificación inicial ..... 26
Figura 4.2: Actividades de alto nivel y diagrama de Gantt de la planificación inicial ..... 27
Figura 4.3: Actividades de la etapa "Estudios iniciales" en la planificación inicial ..... 28
Figura 4.4: Actividades de la etapa "Documentación inicial" en la planificación inicial ..... 28
Figura 4.5: Actividades de la etapa "Planificación y presupuestos iniciales" en la planificación inicial ..... 28
Figura 4.6: Actividades de la etapa "Análisis" en la planificación inicial ..... 29
Figura 4.7: Actividades de la etapa "Diseño" en la planificación inicial ..... 29
Figura 4.8: Actividades de la etapa "Implementación" en la planificación inicial ..... 29
Figura 4.9: Actividades de la etapa "Implementación de las pruebas" en la planificación inicial ..... 30
Figura 4.10: Actividades de la etapa "Despliegue" en la planificación inicial ..... 30
Figura 4.11: Actividades de la etapa "Resto de la documentación" en la planificación inicial ..... 30
Figura 5.1: Diagrama de contexto ..... 37
Figura 5.2: Diagrama de clases de análisis ..... 40
Figura 5.3: Boceto de las zonas de la interfaz de usuario ..... 46
Figura 5.4: Boceto detallado de la interfaz de usuario (mostrando la pestaña Topics-Texto) ..... 47
Figura 5.5: Boceto detallado de la interfaz de usuario (mostrando la pestaña Topics-Nube de palabras) 48
Figura 6.1: Diagrama de Componentes ..... 52
Figura 6.2: Diagrama de Despliegue ..... 53
Figura 6.3: Diagrama de Clases del subsistema Topics and Summary ..... 55
Figura 6.4: Diagrama de Clases del subsistema Web backend ..... 56
Figura 6.5: Diagrama de Clases del subsistema Web frontend ..... 57
Figura 6.6: Prototipo pantalla – Topics en formato texto ..... 58
Figura 6.7: Prototipo pantalla – Topics en formato texto y documentos más representativos ..... 58
Figura 6.8: Prototipo pantalla – Topics en formato nube de palabras ..... 59
Figura 6.9: Prototipo pantalla – Topics en formato nube de palabras y documentos más repr. ..... 59
Figura 6.10: Prototipo pantalla – Configuración de la visualización de topics ..... 60
Figura 6.11: Prototipo pantalla – Obtención de topics relacionados, dado un texto ..... 60
Figura 6.12: Prototipo pantalla – Obtención de documentos similares, dado un texto ..... 61
Figura 6.13: Prototipo pantalla – Obtención de un resumen, dado un texto ..... 61
Figura 8.1: Resultados análisis estático de código (Topics and Summary) ..... 81
Figura 8.2: Resultados análisis estático de código (Web backend) ..... 82
Figura 8.3: Resolución 1440x768 (topics – formato texto) ..... 91
Figura 8.4: Resolución 1440x768 (topics – formato nube de palabras) ..... 91
Figura 8.5: Resolución 1440x768 (topics – configuración) ..... 92
Figura 8.6: Resolución 1440x768 (documentos de topic y sección texto) ..... 92
Figura 8.7: Resolución 1024x768 (texto – prob. topics, documentos similares y resumen) ..... 93
Figura 8.8: Resolución 1024x768 (topics – formato texto) ..... 93
Figura 8.9: Resolución 1024x768 (topics – formato nube de palabras) ..... 94
Figura 8.10: Resolución 1024x768 (topics – configuración) ..... 94
Figura 8.11: Resolución 1024x768 (documentos de topic y sección texto) ..... 95
Figura 8.12: Resolución 1024x768 (texto – prob. topics, documentos similares y resumen) ..... 95
Figura 8.13: Resolución 700x768 (topics – formato texto) ..... 96
Figura 8.14: Resolución 700x768 (topics – formato nube de palabras) ..... 96
Figura 8.15: Resolución 700x768 (topics – configuración) ..... 97
Figura 8.16: Resolución 700x768 (documentos de topic y sección texto) ..... 97

Figura 8.17: Resolución 700x768 (texto – configuración y prob. topics) ..... 98
Figura 8.18: Resolución 700x768 (texto – prob. topics, documentos similares y resumen)..... 98
Figura 8.19: Resolución 320x768 (topics – formato texto) ..... 99
Figura 8.20: Resolución 320x768 (topics – formato nube de palabras)..... 99
Figura 8.21: Resolución 320x768 (topics – configuración) ..... 100
Figura 8.22: Resolución 320x768 (documentos de topic) ..... 100
Figura 8.23: Resolución 320x768 (sección texto) ..... 101
Figura 8.24: Resolución 320x768 (texto – prob. topics) ..... 101
Figura 8.25: Resolución 700x768 (texto –documentos similares)..... 102
Figura 8.26: Resolución 700x768 (texto – resumen) ..... 102
Figura 11.1: Comparativa de duración de las etapas del proyecto en la planificación inicial..... 111
Figura 11.2: Actividades de alto nivel y diagrama de Gantt de la planificación final ..... 112
Figura 11.3: Actividades de la etapa "Estudios iniciales" en la planificación final ..... 112
Figura 11.4: Actividades de la etapa "Documentación inicial" en la planificación final..... 113
Figura 11.5: Actividades de la etapa "Planificación y presupuestos iniciales" en la planificación final ... 113
Figura 11.6: Actividades de la etapa "Análisis" en la planificación final ..... 113
Figura 11.7: Actividades de la etapa "Diseño" en la planificación final..... 114
Figura 11.8: Actividades de la etapa "Implementación" en la planificación final ..... 114
Figura 11.9: Actividades de la etapa "Implementación de las pruebas" en la planificación final..... 115
Figura 11.10: Actividades de la etapa "Resto de la documentación" en la planificación final ..... 115

# Capítulo 1. Memoria del Proyecto

## 1.1 Resumen de la Motivación, Objetivos y Alcance del Proyecto

El presente Trabajo Fin de Grado consiste en el diseño y desarrollo de una aplicación que permita la clasificación de documentos de texto. Para este fin, la aplicación procesa una colección de documentos, que constituyen la base de conocimiento, para:

- **Identificar la temática (topic) de cada grupo de documentos en la colección actual.** Un topic es un conjunto de palabras relacionadas (denominadas keywords), que describen su temática.
- **Identificar la relación de cada documento de la colección con cada uno de los topics.** A cada documento se le asignará una probabilidad de estar relacionado con cada topic.
- **Clasificar cada documento de la colección dentro de un topic.**

Sobre la base del modelo generado, se ofrece al usuario final las funcionalidades de:

- **Dado un nuevo documento de texto, identificar su relación con cada uno de los topics, clasificarlo dentro de uno de ellos, y generar un resumen extractivo.**
- **Identificar los documentos más representativos de cada topic.**
- **Identificar los documentos de la colección más similares a un nuevo documento**.
- **Generar un resumen extractivo de un documento de texto.** Un resumen extractivo consiste en un conjunto de frases extraídas directamente del texto.

La funcionalidad de esta aplicación se ofrece a través de un subsistema que publica una **API REST implementada en Flask**, la cual se comunica directamente con el núcleo de la aplicación descrita previamente, encapsulando su funcionalidad, y que permite su uso programático. Por otro lado, también se ha desarrollado un **frontend implementado en Angular**, que consume dicha API REST, y permite interaccionar con ella a través de una interfaz gráfica.

## 1.2 Resumen de Todos los Aspectos

Este apartado resume el contenido de las secciones posteriores del presente documento:

- **Introducción:** En ella se explica la motivación del proyecto y sus objetivos, y se describe la situación actual, identificando sistemas similares al que se pretende desarrollar, y evaluando distintas alternativas para alcanzar los objetivos propuestos.
- **Aspectos Teóricos:** En esta sección se describen algunos de los conceptos, herramientas y tecnologías que han sido utilizados en el proyecto.
- **Planificación del Proyecto y Presupuesto Iniciales:** En primer lugar, se describen las etapas y actividades en las que se va a dividir la realización del proyecto, indicando su duración, sus dependencias, sus fechas y los recursos encargados de realizarlas. Posteriormente, se muestra el presupuesto de costes resumido, basado en esta planificación.
- **Análisis:** En esta sección se realiza la especificación de requisitos, tanto funcionales como no funcionales, acompañados de los casos de uso. También se detallan los subsistemas en los que se va a dividir el desarrollo de la aplicación, y se definen a alto nivel las clases de las que se va a componer cada subsistema. Por último, se realiza una primera aproximación del diseño de la interfaz de usuario, y se define el plan de pruebas que se va a seguir.
- **Diseño del Sistema:** En esta sección se incluyen los diagramas de componentes, de despliegue y de clases (detallado). También se refina el diseño de la interfaz realizado previamente en la fase de análisis, mediante un prototipo, y se profundiza en qué pruebas se realizarán y cómo, tomando como guía el plan de pruebas.
- **Implementación del Sistema:** Se detallan los estándares y normas seguidos durante la fase de implementación, y los lenguajes de programación, herramientas y programas utilizados. Por último, se describen los problemas encontrados en esta fase y su solución, así como las rutas donde se encuentra la documentación del código fuente.
- **Desarrollo de las Pruebas:** En esta sección se describen los resultados de las pruebas.
- **Manuales del Sistema:** Se indican las rutas donde se encuentran los manuales de instalación, de ejecución y de programador, así como otra documentación adicional.
- **Conclusiones y Ampliaciones:** Se detallan las conclusiones del sistema elaborado, y las ampliaciones que se han contemplado.
- **Planificación del Proyecto y Presupuesto finales:** El contenido de esta sección es análogo al de la sección *Planificación del Proyecto y Presupuesto Iniciales*, pero realizado al terminar el proyecto, para así poder comparar lo que se había planificado inicialmente con lo que ha sucedido en realidad.
- **Referencias Bibliográficas**
- **Apéndices:** En esta sección se incluye un glosario, se describe el contenido entregado en el archivo adjunto, se indica la ruta donde se puede encontrar el código fuente, y se detalla paso a paso como se han calculado los presupuestos.

# Capítulo 2. Introducción

## 2.1 Justificación del Proyecto

La información en forma de texto está en todas partes: correos electrónicos, páginas web, chats, redes sociales, documentos, … Estos textos pueden ser grandes fuentes de información, pero debido a la enorme cantidad de datos de la que se dispone, y a que no tienen un formato estructurado, puede resultar difícil generar valor y extraer conclusiones de ellos.

Una forma de aportar valor dado un conjunto medianamente grande de documentos de texto es mediante su **clasificación** en grupos. Ejemplos de ello son el análisis de sentimientos o la agrupación de noticias en categorías, entre otros.

Hay situaciones en las que los grupos en los que se quiere clasificar los documentos son conocidos, como, por ejemplo, en el análisis de sentimientos, donde se clasifica un texto como positivo o negativo, y en el que se debe disponer de un conjunto de entrenamiento en el que los documentos estén etiquetados con la pertenencia a uno de los dos grupos.

Sin embargo, hay muchas otras situaciones en las que estos grupos son totalmente desconocidos, y los documentos utilizados para entrenar el modelo de aprendizaje automático no están, por tanto, etiquetados con la pertenencia a un determinado grupo. Aquí es donde entra en juego el aprendizaje no supervisado, que permite identificar estos grupos de forma automática, para posteriormente realizar la clasificación sobre ellos.

Este proyecto opta por este segundo enfoque, el del aprendizaje no supervisado, dada su mayor flexibilidad, puesto que la implementación del sistema no se acopla a un conjunto de datos (y grupos) en específico, sino que es el propio sistema el que extrae los grupos, basándose en los documentos que se le proporcionan para el entrenamiento. Por tanto, toda la funcionalidad diseñada puede ser utilizada posteriormente para clasificar otro conjunto de datos distinto al utilizado en el presente proyecto, con ligeros cambios. Esto hace que sus aplicaciones sean muy diversas. Por poner un par de ejemplos, podría utilizarse para organizar en grupos un conjunto de documentos desordenados, o para clasificar un correo electrónico en una determinada carpeta, en base a su contenido. En general, cualquier caso de uso que involucre documentos de texto y su clasificación, es factible de ser resuelto con esta herramienta.

Otra forma de aportar valor dado un documento de texto de gran tamaño es generando un **resumen** de su contenido, y es el segundo gran objetivo de este proyecto. Las aplicaciones de esta funcionalidad también son muy diversas e interesantes. Por ejemplo, un motor de búsqueda podría generar un resumen de un documento para utilizarlo como “vista previa” de su contenido, o una herramienta utilizada en medicina podría resumir información sobre una determinada enfermedad en unas pocas frases. En general, dada la inmensa cantidad de información de la que disponemos, tener la capacidad de sintetizarla y resumirla siempre va a reportar algo positivo.

## 2.2 Objetivos del Proyecto

Los objetivos del presente proyecto son los siguientes:

- Profundizar en las ramas del **Procesamiento de Lenguaje Natural** y el **Aprendizaje no supervisado**.
- Generar un modelo de aprendizaje automático que permita **identificar los topics** presentes en una colección de documentos de texto.
- Utilizar dicho modelo para **clasificar** los documentos de la colección, así como nuevos documentos, dentro de los topics identificados.
- Identificar los **documentos más representativos** de cada topic.
- Identificar los **documentos** de la colección **más similares** a un nuevo documento.
- Generar un modelo de aprendizaje automático que permita obtener **resúmenes extractivos** de documentos de texto.
- Desarrollar una **API REST** que encapsule la funcionalidad mencionada anteriormente.
- Desarrollar una **SPA, utilizando Angular,** que consuma la API REST y permita a los usuarios interaccionar con el sistema de forma sencilla e intuitiva.
- **Desplegar** la aplicación en la nube.
- **Documentar** todo el trabajo realizado, para facilitar su compresión, así como futuras ampliaciones.

## 2.3 Estudio de la Situación Actual

### 2.3.1 Sistemas similares

A continuación, se describen algunos sistemas similares al que se va a desarrollar, comparando lo que ofrecen estos sistemas con lo que se pretende lograr en este proyecto.

#### *2.3.1.1 uClassify*

**Página web:** https://www.uclassify.com

Se trata de una API REST (que acepta formatos JSON y XML) que permite crear y utilizar clasificadores de texto. Dispone de una serie de clasificadores ya creados, con los cuales se puede interactuar a través de la API REST o de una interfaz web muy simple, en la cual se introduce el texto a clasificar. Para generar un clasificador es necesario indicarle las categorías en las que se van a agrupar los documentos, por lo que utiliza aprendizaje supervisado.

##### 2.3.1.1.1 Ventajas de este sistema frente al sistema a desarrollar

- Permite trabajar con textos en inglés, español, francés y sueco
- La API REST acepta formatos JSON y XML
- Dispone de varios clasificadores ya entrenados

##### 2.3.1.1.2 Ventajas del sistema a desarrollar frente a este sistema

- No es necesario indicarle de manera explícita las categorías, sino que es el propio sistema el que infiere la temática de las mismas. Lo único que hay que indicar es el número de categorías a generar. El sistema ofrecerá ciertas métricas que permitirán ajustar de la mejor forma posible el número de categorías óptimo para ese dataset.
- Ofrece las siguientes funcionalidades extra aparte de la clasificación:
    - Identificar los documentos más representativos de cada topic
    - Identificar los documentos de la colección más similares a un nuevo documento
    - Generar resúmenes de documentos de texto
- Dispone de una interfaz web que permite interaccionar con la API REST. En el caso de uClassify, la interfaz web solo permite interaccionar con los clasificadores ya creados, por lo que para nuevos clasificadores sería necesario implementar otra interfaz.

#### *2.3.1.2 MeaningCloud*

**Página web:** https://www.meaningcloud.com/es

Posee diversas APIs REST con distintas funcionalidades de Procesamiento de Lenguaje Natural, como clasificación de texto, detección de idioma, generación de resúmenes, extracción de entidades y conceptos, análisis de sentimiento, …

#### 2.3.1.2.1 Ventajas de este sistema frente al sistema a desarrollar

- Ofrece otras muchas funcionalidades aparte de la clasificación y la generación de resúmenes
- Permite trabajar con textos en un gran número de idiomas, entre los que se incluyen: inglés, español, francés, italiano y portugués

#### 2.3.1.2.2 Ventajas del sistema a desarrollar frente a este sistema

- Ofrece las siguientes funcionalidades, no ofrecidas por MeaningCloud:
    - Identificar los documentos más representativos de cada topic
    - Identificar los documentos de la colección más similares a un nuevo documento
- Dispone de una interfaz web que permite interaccionar con la API REST. En el caso de MeaningCloud, la interfaz web solo permite interaccionar con los ejemplos

### 2.3.2 Evaluación de Alternativas

A continuación, se presentan todas las alternativas planteadas a la implementación del sistema que finalmente se ha llevado a cabo.

#### *2.3.2.1 Generación del modelo de extracción de topics utilizando aprendizaje supervisado*

La idea inicial del proyecto era desarrollar un clasificador de texto utilizando aprendizaje supervisado, prestando especial atención al uso de redes neuronales. Esta idea fue descartada rápidamente, debido a:

- Los pobres resultados ofrecidos por los primeros modelos obtenidos. Estos modelos utilizaban arquitecturas de redes neuronales propuestas en diversos papers enfocados en la clasificación de texto, pero que en la práctica no llegaron a ofrecer los resultados esperados.
- La inflexibilidad del aprendizaje supervisado, que requiere que las categorías en las que se van a agrupar los documentos sean conocidas de antemano, al igual que requiere disponer de un dataset etiquetado con dichas categorías, para poder realizar el entrenamiento.
- El acoplamiento del aprendizaje supervisado al dataset utilizado, dado que se debe enfocar el entrenamiento a agrupar los documentos en las categorías existentes, mientras que, en el caso del aprendizaje no supervisado, es el propio algoritmo el que extrae las categorías basándose en los documentos, por lo que la lógica implementada puede aplicarse más fácilmente a otros datasets.

#### 2.3.2.2 Implementación del backend y el frontend en un único subsistema

Una alternativa a la implementación actual de la página web que permite interactuar con los modelos era implementarlo todo en un mismo subsistema, prescindiendo así de la API REST.

La principal ventaja de esta alternativa era la sencillez de su implementación, dado que:

- El frontend de la aplicación podía realizarse mediante motores de plantillas HTML, evitando así la necesidad de utilizar tecnologías nuevas como Angular o React.
- Ya había implementado sistemas similares.

Pero este tipo de implementación tenía la gran desventaja de que hacía el sistema mucho menos flexible, no permitiendo:

- Desplegar por separado el backend y el frontend.
- Desarrollar nuevos subsistemas reaprovechando la lógica del backend, cosa que sí es posible implementando el backend como una API REST, dado que ésta puede ser consumida por cualquier tipo de aplicación.

#### 2.3.2.3 Implementación del frontend utilizando React

Para la implementación del subsistema *Web frontend* había dos alternativas: Angular y React. Se decidió apostar por Angular por las siguientes cuestiones:

- Utiliza Typescript, el cual facilita el desarrollo y disminuye la probabilidad de cometer errores
- La documentación y el tutorial son mucho más completos
- A nivel personal, considero más interesante aprender Angular que React

#### 2.3.2.4 Despliegue del sistema en Amazon Web Services

Para el despliegue, las dos alternativas principales eran AWS y Heroku.

La principal ventaja de utilizar AWS era que ya tenía conocimiento previo de su utilización. El problema es que la cuenta de la que disponía ya no me permitía crear instancias EC2, por lo que decidí optar por Heroku, dado que:

- Permite desplegar aplicaciones de forma gratuita e indefinida.
- No es necesario ningún tipo de configuración en la nube. Basta con simplemente subir una imagen Docker con todo lo necesario para que la aplicación pueda funcionar.

Pero no todo son ventajas. Heroku tiene los siguientes inconvenientes en su versión gratuita:

- Solo proporciona 512MB de RAM.
- Tras 30 minutos de inactividad, para la máquina sobre la que se ejecuta la aplicación desplegada.

- Corta todas aquellas peticiones realizadas a las aplicaciones desplegadas cuya duración sobrepase los 30 segundos.

Aun así, a pesar de estos inconvenientes, el hecho de no poder desplegar instancias de forma gratuita en AWS, y la posibilidad que da Heroku de mantener la aplicación de forma gratuita e indefinida, fueron factores clave para su elección.

# Capítulo 3. Aspectos Teóricos

En esta sección, se describen brevemente aquellos conceptos, herramientas y tecnologías que han sido utilizados en este proyecto.

## 3.1 NLP

Las siglas NLP significan "*Natural Language Processing*", cuya traducción al español es "*Procesamiento de Lenguaje Natural*". Se trata de una rama de la inteligencia artificial, que engloba una serie de técnicas utilizadas para diseñar algoritmos que permitan a los ordenadores "entender" el lenguaje natural, con el fin de realizar alguna tarea [1].

Algunas de las aplicaciones del Procesamiento de Lenguaje Natural son: reconocimiento del habla, traducción, reconocimiento de entidades, generación de lenguaje natural, extracción de texto de imágenes, análisis de sentimientos, identificación de topics, generación de resúmenes de texto, … [2]

En este proyecto se tocan los dos últimos campos mencionados previamente: la identificación de topics y la generación de resúmenes de texto.

## 3.2 N-gram

Se denomina ngram a un conjunto de n elementos extraídos de un texto (sílabas, letras, palabras, …) que aparecen frecuentemente juntos, y que por separado no aparecen con tanta frecuencia. Por ejemplo, las palabras "United" y "States" es muy probable que formen el bigram[1] "United States" en un conjunto de documentos de texto que hablen de los Estados Unidos, dado que esas 2 palabras seguramente aparezcan más veces juntas que por separado, mientras que las palabras "there" y "is" lo más normal es que no formen un bigram, dado que, aunque puedan aparecer muchas más veces juntas que "United" y "States", también aparecerán muchas veces por separado [3].

Una forma sencilla de dar una puntuación a dos palabras para analizar si podrían formar un bigram es la propuesta en [3]:

$$score(w_i, w_j) = \frac{count(w_i w_j) - \delta}{count(w_i) \cdot count(w_j)}$$

donde $count(w_i w_j)$ representa el número de veces que aparecen seguidas las palabras $w_i$ y $w_j$, $count(w_x)$ representa el número de veces que aparece la palabra $w_x$, y δ es un valor que previene que se formen muchos ngrams consistentes en palabras muy infrecuentes.

[1] Así es como se denominan a los ngrams formados por 2 elementos. Los ngrams formados por 3 elementos se denominan trigrams, y así sucesivamente.

Los pares de palabras cuya puntuación esté por encima del umbral establecido pasarán a formar un bigram. Esto se aplica varias veces a lo largo del conjunto de documentos de texto, por lo que en iteraciones posteriores se podrían juntar un bigram y una palabra suelta, formando así un trigram; y lo mismo podría suceder con dos bigrams, que acabarían formado un 4-gram.

En este proyecto, la generación de ngrams se incluye dentro de la fase de preprocesado de los datasets. Los ngrams generados después del preprocesado serán tratados como una única palabra por el modelo de extracción de topics, por lo que, a priori, la generación de ngrams debería ofrecer mejores resultados en los topics identificados. Esta mejora esperada en los resultados se puede entender con la siguiente explicación:

El ngram “United States” podría ser una de las keywords de un topic cuya temática principal es la geografía. Si no se hubiesen utilizado ngrams, hubieran aparecido en dicho topic las palabras “United” y “States” por separado, dificultando así la interpretación de dicho topic. Además, las apariciones de las palabras “United” y “States” por separado contarían lo mismo que las apariciones de dichas palabras juntas, por lo que “United States” perdería su significado, y aportaría menos a la identificación de los topics y la clasificación de los documentos en ellos.

## 3.3 Word embedding

Se denomina word embedding a un conjunto de modelos, técnicas y algoritmos de Procesamiento de Lenguaje Natural cuyo objetivo es transformar las palabras (o frases en el caso de phrase embedding) en vectores de números reales de baja dimensión [4].

Su principal ventaja es que ofrece una representación eficiente y expresiva, manteniendo la similitud contextual de las palabras, y generando vectores de baja dimensión [5].

Dos métodos muy conocidos de word embedding son Word2Vec [6] y GloVe [7], los cuales se ha visto que son los más eficientes para aprender representaciones vectoriales de las palabras [6], [7].

En [5] se puede encontrar una comparativa de distintos algoritmos para generar word embeddings (LSA, Word2Vec y GloVe) aplicados al campo de la extracción de topics.

En este proyecto, se utiliza word embedding para transformar las frases de un texto en vectores, sobre los cuales aplicar operaciones matemáticas para determinar la similitud entre frases, y posteriormente poder calcular qué frases resumen mejor dicho texto. Esto está explicado con más detalle en la sección 3.5.

## 3.4 LDA

Principal algoritmo utilizado para la extracción de topics de un conjunto de documentos de texto. Una de las utilidades de la extracción de topics, además de la identificación de las temáticas presentes en los documentos, es la clasificación de dichos documentos en estas temáticas.

Las siglas LDA significan "*Latent Diriletch Allocation*", y la idea principal de este algoritmo es la siguiente [8]:

- El algoritmo agrupa las palabras presentes en los documentos en topics, siendo necesario indicarle el número de topics que debe generar
- Cada documento se puede describir mediante una distribución de topics (es decir, cada documento tiene una probabilidad de pertenecer a cada uno de los topics identificados)
- Cada topic se puede describir mediante una distribución de palabras (es decir, cada topic tiene una probabilidad de estar relacionado con cada una de las palabras presentes en el conjunto de documentos de texto, después de aplicarles el preprocesado)
- Las palabras presentes en cada documento son las que definen su probabilidad de pertenencia a los topics
- No tiene en cuenta el orden de las palabras en los documentos (sigue el modelo "bag of words" o "bolsa de palabras")

En [9] se realiza una explicación mucho más detallada de este algoritmo.

En este proyecto, se han utilizado las implementaciones de LDA tanto de la librería Gensim [10] como de la librería Mallet [11], siendo la implementación de esta última la que mejores resultados ha dado.

## 3.5 TextRank

Algoritmo utilizado para generar resúmenes de documentos de texto. Está basado en el algoritmo PageRank, utilizado por Google para asignar una puntuación numérica a las páginas web (o a los documentos de texto) indexados por un motor de búsqueda, en función de su importancia [12]. Los pasos que sigue el algoritmo TextRank para resumir un texto son los siguientes [13]:

1. Dividir el texto en frases
2. Preprocesar cada frase
3. Transformar cada frase en un vector usando un modelo de word embedding. Para ello:
    a. Se transforma cada palabra de la frase en un vector, usando un modelo preentrenado de word embedding
    b. Se calcula la media de los vectores de las palabras de la frase
4. Obtener la similitud entre las frases del texto. Para ello:
    a. Se realiza una matriz triangular en la que las filas y las columnas son las frases
    b. Se rellena dicha matriz con los valores de la similitud entre las frases, utilizando para ello la similitud del coseno de los vectores de las frases
5. Transformar la matriz de similitud en un grafo, donde los nodos son las frases, y los arcos los valores de similitud entre las frases
6. Aplicar PageRank al grafo anterior, hasta que las puntuaciones de las frases converjan
7. Ordenar las frases por la puntuación dada por PageRank

Más información acerca de este algoritmo se puede encontrar en [14].

## 3.6 Gensim

Librería Python de aprendizaje automático, enfocada en la agrupación de textos en clusters y en el análisis automático de topics presentes en documentos de texto [10].

**Creador:** Andrew McCallum

**Página web:** https://radimrehurek.com/gensim/

**Versión utilizada:** 3.4.0

Ha sido utilizada para generar y utilizar modelos de extracción de topics en documentos de texto. Implementa un wrapper que permite utilizar el algoritmo LDA implementado en la librería Mallet desde esta misma librería.

## 3.7 Mallet

Librería Java de aprendizaje automático, enfocada en el Procesamiento de Lenguaje Natural, la clasificación de texto, la identificación de topics y otros campos relacionados con el procesamiento de texto [11].

**Creador:** Andrew McCallum

**Página web:** http://mallet.cs.umass.edu

**Versión utilizada:** 2.0.8

Ha sido utilizada de forma indirecta, a través de un wrapper de la librería gensim, para generar y utilizar el modelo de extracción de topics de documentos de texto denominado LDA Mallet.

## 3.8 Flask

Microframework para desarrollo web en el lenguaje de programación Python.

**Creador:** Armin Ronacher

**Página web:** http://flask.pocoo.org/docs/1.0/

**Versión utilizada:** 1.0.2

Ha sido utilizado para implementar el backend, que ofrece una API REST con la funcionalidad de los modelos.

## 3.9 REST

Las siglas REST significan "*Representational State Transfer*". Se trata de un estilo de arquitectura software para sistemas hipermedia distribuidos [15]. En REST, los datos y la funcionalidad se consideran recursos, y se accede a ellos utilizando URIs. No es un estándar, pero está basado en estándares, como HTTP, URI, XML, … El formato de los datos que se transmiten a través de REST no tiene un formato definido, por lo que puede ser texto normal, JSON, XML, … Es una alternativa a otros protocolos de intercambio de datos como SOAP.

Su principal ventaja es que permite la interoperabilidad entre sistemas, es decir, permite comunicar sistemas implementados en distintos lenguajes, con distintos frameworks, librerías, …

**Creador:** Roy Thomas Fielding

Ha sido utilizado para comunicar el backend y el frontend, utilizando JSON como formato para el envío de los datos.

## 3.10 Angular

Framework para aplicaciones web desarrollado en TypeScript, y mantenido por Google, que permite desarrollar aplicaciones del tipo SPA (Single Page Application).

**Página web:** https://angular.io

**Versión utilizada:** 7.2.0

Ha sido utilizado para implementar el frontend.

## 3.11 SPA

Las siglas SPA significan "*Single Page Application*", cuya traducción al español es "Aplicación de página única". Se trata de un tipo de aplicación web en la que todos los contenidos de la misma se sirven en una única página. En lugar de cargar nuevas páginas del servidor, los contenidos de la página se van actualizando dinámicamente, lo cual favorece la experiencia de usuario, dado que permite interactuar con ella de forma más fluida [16].

Al haber implementado el frontend utilizando el framework Angular, la aplicación web resultante se trata de una SPA.

# Capítulo 4. Planificación del Proyecto y Presupuesto Iniciales

## 4.1 Planificación Inicial

El proyecto tiene una duración total estimada de **300 horas**. La fecha de **comienzo** es el **1 de febrero de 2019**, y la fecha de **finalización** estimada es el **11 de junio de 2019**.

El proyecto se ha dividido en las siguientes 9 etapas:

| Etapa | Duración | Fecha de comienzo | Fecha de fin |
|---|---|---|---|
| **Estudios iniciales** | 25 horas | 01/02/2019 | 11/02/2019 |
| **Documentación inicial** | 3 horas | 11/02/2019 | 12/02/2019 |
| **Planificación y presupuestos iniciales** | 11 horas | 13/02/2019 | 17/02/2019 |
| **Análisis** | 20 horas | 17/02/2019 | 26/02/2019 |
| **Diseño** | 27 horas | 26/02/2019 | 10/03/2019 |
| **Implementación** | 124 horas | 10/03/2019 | 03/05/2019 |
| **Implementación de las pruebas** | 32 horas | 04/05/2019 | 17/05/2019 |
| **Despliegue** | 8 horas | 18/05/2019 | 20/05/2019 |
| **Resto de la documentación** | 50 horas | 21/05/2019 | 11/06/2019 |

A continuación, se muestra un gráfico circular que compara la duración de las etapas mencionadas anteriormente:

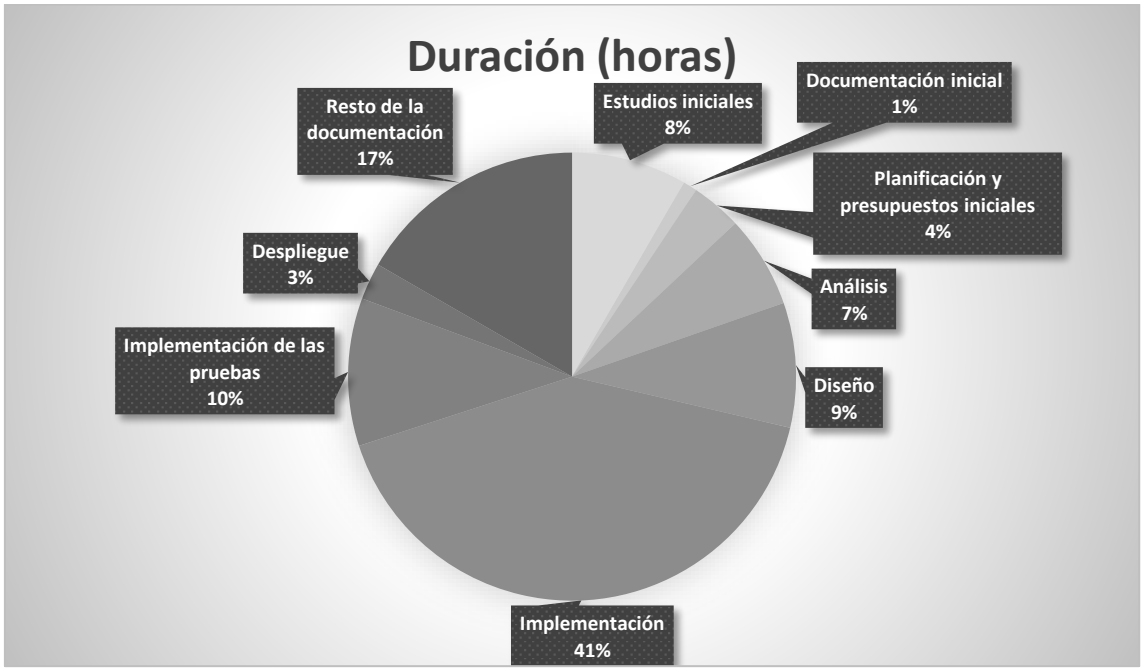



*Figura 4.1: Comparativa de duración de las etapas del proyecto en la planificación inicial*

Para la planificación se ha tenido en cuenta una jornada laboral de **16 horas a la semana**.

Cabe mencionar los siguientes detalles sobre esta planificación inicial:

- Los estudios iniciales han sido realizados antes que la planificación, dado que son necesarios para establecer el alcance de proyecto y poder realizar el WBS. Dichos estudios han sido añadidos dentro de la planificación, dado que las horas dedicadas a ellos también computan dentro de las 300 horas estimadas para la realización del Trabajo Fin de Grado.
- La duración de los estudios iniciales y la duración de la realización de la planificación inicial no han sido estimadas, sino que se han indicado los tiempos reales que se han tardado en realizar dichas actividades, dado que su duración ya era conocida.
- Las pruebas unitarias no se han incluido dentro de la implementación, debido a que, como el Plan de Pruebas está aún sin definir, no se sabe en qué partes del software se van a realizar pruebas unitarias automatizadas. Lo que sí está claro es que serán necesarias pruebas unitarias en algunas partes del sistema, por lo que se han añadido como una actividad aparte de la implementación.

Para realizar la planificación se ha utilizado la herramienta Microsoft Project. A continuación, se muestra una imagen con las actividades de más alto nivel (etapas) y los hitos, así como un diagrama de Gantt:

| | Núm de esqu | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras |
|---|---|---|---|---|---|---|
| 1 | 1 | **Realización del TFG** | **300 horas** | **vie 01/02/19** | **mar 11/06/19** | |
| 2 | 1.1 | Inicio | 0 horas | vie 01/02/19 | vie 01/02/19 | |
| 3 | 1.2 | **Estudios iniciales** | **25 horas** | **vie 01/02/19** | **lun 11/02/19** | **2** |
| 11 | 1.3 | Terminados los estudios iniciales | 0 horas | lun 11/02/19 | lun 11/02/19 | 3 |
| 12 | 1.4 | **Documentación inicial** | **3 horas** | **lun 11/02/19** | **mar 12/02/19** | **11** |
| 17 | 1.5 | Terminada la documentación inicial | 0 horas | mar 12/02/19 | mar 12/02/19 | 12 |
| 18 | 1.6 | **Planificación y presupuestos iniciales** | **11 horas** | **mié 13/02/19** | **dom 17/02/19** | **17** |
| 25 | 1.7 | Terminada la planificación y los presupuestos iniciales | 0 horas | dom 17/02/19 | dom 17/02/19 | 18 |
| 26 | 1.8 | **Análisis** | **20 horas** | **dom 17/02/19** | **mar 26/02/19** | **25** |
| 39 | 1.9 | Terminado el análisis | 0 horas | mar 26/02/19 | mar 26/02/19 | 26 |
| 40 | 1.10 | **Diseño** | **27 horas** | **mar 26/02/19** | **dom 10/03/19** | **39** |
| 50 | 1.11 | Terminado el diseño | 0 horas | dom 10/03/19 | dom 10/03/19 | 40 |
| 51 | 1.12 | **Implementación** | **124 horas** | **dom 10/03/19** | **vie 03/05/19** | **50** |
| 61 | 1.13 | Terminada la implementación | 0 horas | vie 03/05/19 | vie 03/05/19 | 51 |
| 62 | 1.14 | **Implementación de las pruebas** | **32 horas** | **sáb 04/05/19** | **vie 17/05/19** | **61** |
| 65 | 1.15 | Terminado el desarrollo de las pruebas | 0 horas | vie 17/05/19 | vie 17/05/19 | 62 |
| 66 | 1.16 | **Despliegue** | **8 horas** | **sáb 18/05/19** | **lun 20/05/19** | **65** |
| 69 | 1.17 | Terminado el despliegue | 0 horas | lun 20/05/19 | lun 20/05/19 | 66 |
| 70 | 1.18 | **Resto de la documentación** | **50 horas** | **mar 21/05/19** | **mar 11/06/19** | **69** |
| 103 | 1.19 | Terminada el resto de la documentación | 0 horas | mar 11/06/19 | mar 11/06/19 | 70 |
| 104 | 2 | Terminado el TFG | 0 horas | mar 11/06/19 | mar 11/06/19 | 1 |

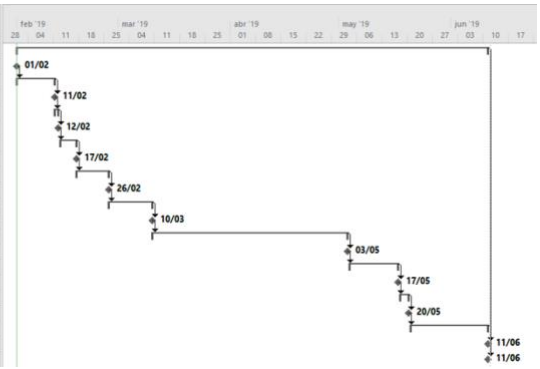



*Figura 4.2: Actividades de alto nivel y diagrama de Gantt de la planificación inicial*

Las filas en azul son las actividades de más alto nivel, y las filas en verde son los hitos.

A continuación, se muestra el WBS detallado[2], con las actividades de cada una de las etapas:

[2] El WBS completo se encuentra en el archivo adjunto *Planificación inicial.mpp.*

### 4.1.1 Estudios iniciales

| | Núm de esquı | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 3 | **1.2** | **Estudios iniciales** | **25 horas** | **vie 01/02/19** | **lun 11/02/19** | **2** | | **25 horas** | **No** |
| 4 | 1.2.1 | Definición de los objetivos del proyecto | 3 horas | vie 01/02/19 | sáb 02/02/19 | | Desarrollador fullstack | 3 horas | No |
| 5 | 1.2.2 | Formación en Sistemas de Recuperación de información y NLP | 6 horas | sáb 02/02/19 | lun 04/02/19 | 4 | Desarrollador fullstack | 6 horas | No |
| 6 | 1.2.3 | Lectura de papers sobre modelos para la extracción de topics y clasificación de texto | 8 horas | lun 04/02/19 | vie 08/02/19 | 5 | Desarrollador fullstack | 8 horas | No |
| 7 | 1.2.4 | Lectura de papers sobre modelos para la generación de resumenes de textos | 2 horas | vie 08/02/19 | sáb 09/02/19 | 6 | Desarrollador fullstack | 2 horas | No |
| 8 | 1.2.5 | Búsqueda y selección del dataset | 3 horas | sáb 09/02/19 | dom 10/02/19 | 7 | Desarrollador fullstack | 3 horas | No |
| 9 | 1.2.6 | Investigación de librerías para la generación de los modelos de aprendizaje automático | 1 hora | dom 10/02/19 | dom 10/02/19 | 8 | Desarrollador fullstack | 1 hora | No |
| 10 | 1.2.7 | Comparación y selección de los lenguajes, herramientas, librerías y frameworks a utilizar para implementar el software del proyecto | 2 horas | dom 10/02/19 | lun 11/02/19 | 9 | Desarrollador fullstack | 2 horas | No |
| 11 | 1.3 | Terminados los estudios iniciales | 0 horas | lun 11/02/19 | lun 11/02/19 | 3 | | 0 horas | Sí |

*Figura 4.3: Actividades de la etapa "Estudios iniciales" en la planificación inicial*

### 4.1.2 Documentación inicial

| | Núm de esquı | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 12 | **1.4** | **Documentación inicial** | **3 horas** | **lun 11/02/19** | **mar 12/02/19** | **11** | | **3 horas** | **No** |
| 13 | **1.4.1** | **Introducción** | **3 horas** | **lun 11/02/19** | **mar 12/02/19** | | | **3 horas** | **No** |
| 14 | 1.4.1.1 | Justificación del proyecto | 1 hora | lun 11/02/19 | lun 11/02/19 | | Desarrollador fullstack | 1 hora | No |
| 15 | 1.4.1.2 | Objetivos del proyecto | 0,5 horas | mar 12/02/19 | mar 12/02/19 | 14 | Desarrollador fullstack | 0,5 horas | No |
| 16 | 1.4.1.3 | Estudio de la situación actual | 1,5 horas | mar 12/02/19 | mar 12/02/19 | 15 | Desarrollador fullstack | 1,5 horas | No |
| 17 | 1.5 | Terminada la documentación inicial | 0 horas | mar 12/02/19 | mar 12/02/19 | 12 | | 0 horas | Sí |

*Figura 4.4: Actividades de la etapa "Documentación inicial" en la planificación inicial*

### 4.1.3 Planificación y presupuestos iniciales

| | Núm de esquı | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 18 | **1.6** | **Planificación y presupuestos iniciales** | **11 horas** | **mié 13/02/19** | **dom 17/02/19** | **17** | | **11 horas** | **No** |
| 19 | **1.6.1** | **Planificación inicial** | **5 horas** | **mié 13/02/19** | **vie 15/02/19** | | | **5 horas** | **No** |
| 20 | 1.6.1.1 | Creación de la Estructura de Desglose de Trabajo (WBS) | 4 horas | mié 13/02/19 | jue 14/02/19 | | Desarrollador fullstack | 4 horas | No |
| 21 | 1.6.1.2 | Estimación de actividades | 1 hora | vie 15/02/19 | vie 15/02/19 | 20 | Desarrollador fullstack | 1 hora | No |
| 22 | **1.6.2** | **Presupuesto inicial** | **6 horas** | **vie 15/02/19** | **dom 17/02/19** | **19** | | **6 horas** | **No** |
| 23 | 1.6.2.1 | Presupuesto de costes inicial | 4 horas | vie 15/02/19 | sáb 16/02/19 | | Desarrollador fullstack | 4 horas | No |
| 24 | 1.6.2.2 | Presupuesto de cliente inicial | 2 horas | dom 17/02/19 | dom 17/02/19 | 23 | Desarrollador fullstack | 2 horas | No |
| 25 | 1.7 | Terminada la planificación y los presupuestos iniciales | 0 horas | dom 17/02/19 | dom 17/02/19 | 18 | | 0 horas | Sí |

*Figura 4.5: Actividades de la etapa "Planificación y presupuestos iniciales" en la planificación inicial*

### 4.1.4 Análisis

| | Núm de esquı | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 26 | **1.8** | **Análisis** | **20 horas** | **dom 17/02/19** | **mar 26/02/19** | **25** | | **20 horas** | **No** |
| 27 | 1.8.1 | Definición del alcance del sistema | 1 hora | dom 17/02/19 | dom 17/02/19 | | Desarrollador fullstack | 1 hora | No |
| 28 | **1.8.2** | **Requisitos** | **4 horas** | **lun 18/02/19** | **mar 19/02/19** | **27** | | **4 horas** | **No** |
| 29 | 1.8.2.1 | Obtención de requisitos | 1 hora | lun 18/02/19 | lun 18/02/19 | | Desarrollador fullstack | 1 hora | No |
| 30 | 1.8.2.2 | Especificación de requisitos | 3 horas | lun 18/02/19 | mar 19/02/19 | 29 | Desarrollador fullstack | 3 horas | No |
| 31 | 1.8.3 | Casos de Uso | 2 horas | mié 20/02/19 | mié 20/02/19 | 28 | Desarrollador fullstack | 2 horas | No |
| 32 | 1.8.4 | Identificación de subsistemas | 0,5 horas | jue 21/02/19 | jue 21/02/19 | 31 | Desarrollador fullstack | 0,5 horas | No |
| 33 | 1.8.5 | Diagrama de clases preliminar del análisis | 6 horas | jue 21/02/19 | sáb 23/02/19 | 32 | Desarrollador fullstack | 6 horas | No |
| 34 | **1.8.6** | **Análisis de interfaces de usuario** | **5,5 horas** | **sáb 23/02/19** | **lun 25/02/19** | **33** | | **5,5 horas** | **No** |
| 35 | 1.8.6.1 | Prototipos de pantallas a papel | 2 horas | sáb 23/02/19 | dom 24/02/19 | | Desarrollador fullstack | 2 horas | No |
| 36 | 1.8.6.2 | Prototipos de pantallas con herramienta | 3 horas | dom 24/02/19 | lun 25/02/19 | 35 | Desarrollador fullstack | 3 horas | No |
| 37 | 1.8.6.3 | Diagrama de navegabilidad | 0,5 horas | lun 25/02/19 | lun 25/02/19 | 36 | Desarrollador fullstack | 0,5 horas | No |
| 38 | 1.8.7 | Especificación del plan de pruebas | 1 hora | mar 26/02/19 | mar 26/02/19 | 34 | Desarrollador fullstack | 1 hora | No |
| 39 | 1.9 | Terminado el análisis | 0 horas | mar 26/02/19 | mar 26/02/19 | 26 | | 0 horas | Sí |

*Figura 4.6: Actividades de la etapa "Análisis" en la planificación inicial*

### 4.1.5 Diseño

| | Núm de esquı | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 40 | **1.10** | **Diseño** | **27 horas** | **mar 26/02/19** | **dom 10/03/19** | **39** | | **27 horas** | **No** |
| 41 | **1.10.1** | **Arquitectura del sistema** | **7 horas** | **mar 26/02/19** | **vie 01/03/19** | | | **7 horas** | **No** |
| 42 | 1.10.1.1 | Diagrama de paquetes | 3 horas | mar 26/02/19 | mié 27/02/19 | | Desarrollador fullstack | 3 horas | No |
| 43 | 1.10.1.2 | Diagrama de componentes | 3 horas | jue 28/02/19 | vie 01/03/19 | 42 | Desarrollador fullstack | 3 horas | No |
| 44 | 1.10.1.3 | Diagramas de despliegue | 1 hora | vie 01/03/19 | vie 01/03/19 | 43 | Desarrollador fullstack | 1 hora | No |
| 45 | **1.10.2** | **Diseño de clases** | **12 horas** | **sáb 02/03/19** | **mié 06/03/19** | **41** | | **12 horas** | **No** |
| 46 | 1.10.2.1 | Identificación de las clases, sus detalles y sus relaciones | 10 horas | sáb 02/03/19 | mar 05/03/19 | | Desarrollador fullstack | 10 horas | No |
| 47 | 1.10.2.2 | Diagrama de clases | 2 horas | mié 06/03/19 | mié 06/03/19 | 46 | Desarrollador fullstack | 2 horas | No |
| 48 | 1.10.3 | Diseño de la interfaz | 2 horas | jue 07/03/19 | jue 07/03/19 | 45 | Desarrollador fullstack | 2 horas | No |
| 49 | 1.10.4 | Especificación técnica del plan de pruebas | 6 horas | vie 08/03/19 | dom 10/03/19 | 48 | Desarrollador fullstack | 6 horas | No |
| 50 | 1.11 | Terminado el diseño | 0 horas | dom 10/03/19 | dom 10/03/19 | 40 | | 0 horas | Sí |

*Figura 4.7: Actividades de la etapa "Diseño" en la planificación inicial*

### 4.1.6 Implementación

| | Núm de esquı | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 51 | **1.12** | **Implementación** | **124 horas** | **dom 10/03/19** | **vie 03/05/19** | **50** | | **124 horas** | **No** |
| 52 | 1.12.1 | Formación en tecnologías Python y librerías Data science/NLP | 4 horas | dom 10/03/19 | lun 11/03/19 | | Desarrollador fullstack | 4 horas | No |
| 53 | **1.12.2** | **Implementación del subsistema que contiene los modelos de aprendizaje automático** | **80 horas** | **mar 12/03/19** | **lun 15/04/19** | **52** | | **80 horas** | **No** |
| 54 | 1.12.2.1 | Preparación y preprocesado del dataset | 30 horas | mar 12/03/19 | dom 24/03/19 | | Desarrollador fullstack | 30 horas | No |
| 55 | 1.12.2.2 | Impementación de modelos de aprendizaje automático para la extracción de topics y clasificación de texto | 40 horas | lun 25/03/19 | jue 11/04/19 | 54 | Desarrollador fullstack | 40 horas | No |
| 56 | 1.12.2.3 | Implementación de modelos de aprendizaje automático para la generación de resumenes de textos | 10 horas | vie 12/04/19 | lun 15/04/19 | 55 | Desarrollador fullstack | 10 horas | No |
| 57 | 1.12.3 | Formación en Flask | 2 horas | mar 16/04/19 | mar 16/04/19 | 53 | Desarrollador fullstack | 2 horas | No |
| 58 | 1.12.4 | Implementación del subsistema que contiene el backend de la parte web | 10 horas | mié 17/04/19 | dom 21/04/19 | 57 | Desarrollador fullstack | 10 horas | No |
| 59 | 1.12.5 | Formación en Angular | 8 horas | dom 21/04/19 | mié 24/04/19 | 58 | Desarrollador fullstack | 8 horas | No |
| 60 | 1.12.6 | Implementación del subsistema que contiene el frontend de la parte web | 20 horas | jue 25/04/19 | vie 03/05/19 | 59 | Desarrollador fullstack | 20 horas | No |
| 61 | 1.13 | Terminada la implementación | 0 horas | vie 03/05/19 | vie 03/05/19 | 51 | | 0 horas | Sí |

*Figura 4.8: Actividades de la etapa "Implementación" en la planificación inicial*

### 4.1.7 Implementación de las pruebas

| | Núm de esque | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 62 | **1.14** | **Implementación de las pruebas** | **32 horas** | **sáb 04/05/19** | **vie 17/05/19** | **61** | | **32 horas** | **No** |
| 63 | 1.14.1 | Pruebas unitarias | 12 horas | sáb 04/05/19 | mié 08/05/19 | | Desarrollador fullstack | 12 horas | No |
| 64 | 1.14.2 | Otras pruebas detalladas en el Plan de Pruebas | 20 horas | jue 09/05/19 | vie 17/05/19 | 63 | Desarrollador fullstack | 20 horas | No |
| 65 | 1.15 | Terminada la implementación de las pruebas | 0 horas | vie 17/05/19 | vie 17/05/19 | 62 | | 0 horas | Sí |

*Figura 4.9: Actividades de la etapa "Implementación de las pruebas" en la planificación inicial*

### 4.1.8 Despliegue

| | Núm de esque | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 66 | **1.16** | **Despliegue** | **8 horas** | **sáb 18/05/19** | **lun 20/05/19** | **65** | | **8 horas** | **No** |
| 67 | 1.16.1 | Comparar y seleccionar la plataforma de despliegue en la nube | 1 hora | sáb 18/05/19 | sáb 18/05/19 | | Desarrollador fullstack | 1 hora | No |
| 68 | 1.16.2 | Realizar el despliegue en la plataforma seleccionada | 7 horas | sáb 18/05/19 | lun 20/05/19 | 67 | Desarrollador fullstack | 7 horas | No |
| 69 | 1.17 | Terminado el despliegue | 0 horas | lun 20/05/19 | lun 20/05/19 | 66 | | 0 horas | Sí |

*Figura 4.10: Actividades de la etapa "Despliegue" en la planificación inicial*

### 4.1.9 Resto de la documentación

| | Núm de esque | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 70 | **1.18** | **Resto de la documentación** | **50 horas** | **mar 21/05/19** | **mar 11/06/19** | **69** | | **50 horas** | **No** |
| 71 | **1.18.1** | **Documentación de la implementación del sistema** | **3,5 horas** | **mar 21/05/19** | **mié 22/05/19** | | | **3,5 horas** | **No** |
| 72 | 1.18.1.1 | Estándares y Normas seguidos | 0,5 horas | mar 21/05/19 | mar 21/05/19 | | Desarrollador fullstack | 0,5 horas | No |
| 73 | 1.18.1.2 | Lenguajes de programación | 0,5 horas | mar 21/05/19 | mar 21/05/19 | 72 | Desarrollador fullstack | 0,5 horas | No |
| 74 | 1.18.1.3 | Herramientas y Programas usados para el desarrollo | 1 hora | mar 21/05/19 | mar 21/05/19 | 73 | Desarrollador fullstack | 1 hora | No |
| 75 | 1.18.1.4 | Creación del sistema | 1,5 horas | mié 22/05/19 | mié 22/05/19 | 74 | Desarrollador fullstack | 1,5 horas | No |
| 76 | **1.18.2** | **Documentación del desarrollo de las pruebas** | **8 horas** | **mié 22/05/19** | **dom 26/05/19** | **71** | | **8 horas** | **No** |
| 77 | 1.18.2.1 | Pruebas unitarias | 2 horas | mié 22/05/19 | jue 23/05/19 | | Desarrollador fullstack | 2 horas | No |
| 78 | 1.18.2.2 | Otras pruebas detalladas en el Plan de Pruebas | 6 horas | jue 23/05/19 | dom 26/05/19 | 77 | Desarrollador fullstack | 6 horas | No |
| 79 | **1.18.3** | **Manuales del sistema** | **11,5 horas** | **dom 26/05/19** | **vie 31/05/19** | **76** | | **11,5 horas** | **No** |
| 80 | 1.18.3.1 | Manual de instalación | 2,5 horas | dom 26/05/19 | dom 26/05/19 | | Desarrollador fullstack | 2,5 horas | No |
| 81 | 1.18.3.2 | Manual de ejecución | 2 horas | lun 27/05/19 | lun 27/05/19 | 80 | Desarrollador fullstack | 2 horas | No |
| 82 | 1.18.3.3 | Manual de usuario | 4 horas | mar 28/05/19 | mié 29/05/19 | 81 | Desarrollador fullstack | 4 horas | No |
| 83 | 1.18.3.4 | Manual del programador | 3 horas | jue 30/05/19 | vie 31/05/19 | 82 | Desarrollador fullstack | 3 horas | No |
| 84 | 1.18.4 | Aspectos teóricos | 2 horas | vie 31/05/19 | sáb 01/06/19 | 79 | Desarrollador fullstack | 2 horas | No |
| 85 | **1.18.5** | **Conclusiones y ampliaciones** | **3,5 horas** | **sáb 01/06/19** | **dom 02/06/19** | **84** | | **3,5 horas** | **No** |
| 86 | 1.18.5.1 | Conclusiones | 2,5 horas | sáb 01/06/19 | dom 02/06/19 | | Desarrollador fullstack | 2,5 horas | No |
| 87 | 1.18.5.2 | Ampliaciones | 1 hora | dom 02/06/19 | dom 02/06/19 | 86 | Desarrollador fullstack | 1 hora | No |
| 88 | 1.18.6 | Memoria del proyecto | 4,5 horas | dom 02/06/19 | mar 04/06/19 | 85 | Desarrollador fullstack | 4,5 horas | No |
| 89 | 1.18.7 | Referencias bibliográficas | 1,5 horas | mar 04/06/19 | mié 05/06/19 | 88 | Desarrollador fullstack | 1,5 horas | No |
| 90 | **1.18.8** | **Apéndices** | **8,5 horas** | **mié 05/06/19** | **sáb 08/06/19** | **89** | | **8,5 horas** | **No** |
| 91 | 1.18.8.1 | Glosario | 1,5 horas | mié 05/06/19 | mié 05/06/19 | | Desarrollador fullstack | 1,5 horas | No |
| 92 | 1.18.8.2 | Contenido entregado en el archivo adjunto | 3,5 horas | jue 06/06/19 | vie 07/06/19 | 91 | Desarrollador fullstack | 3,5 horas | No |
| 93 | 1.18.8.3 | Índice alfabético | 1 hora | vie 07/06/19 | sáb 08/06/19 | 92 | Desarrollador fullstack | 1 hora | No |
| 94 | 1.18.8.4 | Código fuente | 2,5 horas | sáb 08/06/19 | sáb 08/06/19 | 93 | Desarrollador fullstack | 2,5 horas | No |
| 95 | 1.18.9 | Agradecimientos, resumen y keywords | 1 hora | dom 09/06/19 | dom 09/06/19 | 90 | Desarrollador fullstack | 1 hora | No |
| 96 | **1.18.10** | **Planificación y presupuestos finales** | **6 horas** | **dom 09/06/19** | **mar 11/06/19** | **95** | | **6 horas** | **No** |
| 97 | **1.18.10.** | **Planificación final** | **3 horas** | **dom 09/06/19** | **lun 10/06/19** | | | **3 horas** | **No** |
| 98 | 1.18.10. | Modificación de la Estructura de Desglose de Trabajo (WBS) | 2 horas | dom 09/06/19 | dom 09/06/19 | | Desarrollador fullstack | 2 horas | No |
| 99 | 1.18.10. | Actualización de las estimaciones de las actividades con los tiempos reales | 1 hora | lun 10/06/19 | lun 10/06/19 | 98 | Desarrollador fullstack | 1 hora | No |
| 100 | **1.18.10.** | **Presupuesto final** | **3 horas** | **lun 10/06/19** | **mar 11/06/19** | **97** | | **3 horas** | **No** |
| 101 | 1.18.10. | Presupuesto de costes final | 2 horas | lun 10/06/19 | mar 11/06/19 | | Desarrollador fullstack | 2 horas | No |
| 102 | 1.18.10. | Presupuesto de cliente final | 1 hora | mar 11/06/19 | mar 11/06/19 | 101 | Desarrollador fullstack | 1 hora | No |
| 103 | 1.19 | Terminada el resto de la documentación | 0 horas | mar 11/06/19 | mar 11/06/19 | 70 | | 0 horas | Sí |
| 104 | 2 | Terminado el TFG | 0 horas | mar 11/06/19 | mar 11/06/19 | 1 | | 0 horas | Sí |

*Figura 4.11: Actividades de la etapa "Resto de la documentación" en la planificación inicial*

## 4.2 Presupuesto de Costes Inicial (resumido)

A continuación, se muestra el presupuesto de costes resumido, basado en la planificación anterior:

| **Presupuesto de costes** | | |
|---|---|---|
| ***Cód.*** | ***Partida*** | ***Total*** |
| 01 | Estudios iniciales y planificación inicial | 1.350,00 € |
| 02 | Análisis y diseño del sistema | 1.762,50 € |
| 03 | Implementación y despliegue del sistema | 6.248,40 € |
| 04 | Documentación | 1.987,50 € |
| 05 | Otros costes | 1.134,60 € |
| | **Total Coste** | **12.483,00 €** |

*Tabla 1: Presupuesto de costes inicial (resumido)*

**Nota:** El coste que figura aquí no incluye el IVA.

En el apéndice *Presupuesto de Costes Inicial (detallado)* se detalla cómo se ha calculado este presupuesto de costes inicial.

# Capítulo 5. Análisis

## 5.1 Definición del Sistema

### 5.1.1 Determinación del Alcance del Sistema

El alcance del sistema cubre todo lo descrito en la sección *Resumen de la Motivación, Objetivos y Alcance del Proyecto*.

En esta fase de análisis se hace referencia, además, a un segundo actor, aparte del usuario normal, que es el **administrador del sistema**, cuyas funciones principales son consultar y modificar los parámetros del sistema. Respecto a este nuevo actor, solamente se van a llegar a especificar sus funcionalidades hasta el diagrama de clases del análisis, por lo que no se va a especificar la interfaz del administrador, ni se va a tener en cuenta para las etapas de diseño, implementación y pruebas. Esto se deja como futura ampliación del sistema.

## 5.2 Requisitos del Sistema

### 5.2.1 Obtención de los Requisitos del Sistema

#### *5.2.1.1 Requisitos de interfaces externas*

##### 5.2.1.1.1 Interfaces de usuario

**RIE-IU-1.** El sistema será accesible a los usuarios a partir de las siguientes versiones de los siguientes navegadores web:

**RIE-IU-1.1.** Google Chrome.

**RIE-IU-1.1.1.** Versión 71.0.3578.80.

**RIE-IU-1.2.** Mozilla Firefox.

**RIE-IU-1.2.1.** Versión 63.0.3.

**RIE-IU-1.3.** Opera.

**RIE-IU-1.3.1.** Versión 57.0.3098.91.

**RIE-IU-1.4.** Safari.

**RIE-IU-1.4.1.** Versión 12.0.1.

**RIE-IU-1.5.** Microsoft Edge.

**RIE-IU-1.5.1.** Versión 42.17134.1.0.

#### *5.2.1.2 Requisitos funcionales*

**RF-1.** El sistema generará un modelo de extracción de topics basado en un conjunto de documentos de texto.

**RF-1.1.** El idioma de los documentos será el inglés.

**RF-2.** El sistema generará un modelo que permita realizar resúmenes extractivos de documentos de texto.

**RF-2.1.** El idioma de los documentos será el inglés.

**RF-3.** El sistema permitirá al usuario visualizar los topics del modelo entrenado.

**RF-3.1.** Los topics se podrán visualizar en formato textual.

**RF-3.1.1.** Por cada topic se mostrarán las NUM_KEYWORDS_TOPIC_TEXTO keywords más relevantes de dicho topic.

**RF-3.1.1.1.** El valor por defecto de NUM_KEYWORDS_TOPIC_TEXTO es de 5.

**RF-3.1.1.2.** NUM_KEYWORDS_TOPIC_TEXTO debe poder ser configurado por el usuario.

**RF-3.1.1.2.1.** El sistema verificará que el valor de NUM_KEYWORDS_TOPIC_TEXTO sea mayor de 0.

**RF-3.1.1.2.2.** El sistema verificará que el valor de NUM_KEYWORDS_TOPIC_TEXTO sea menor o igual que MAX_NUM_KEYWORDS_TOPIC_TEXTO.

**RF-3.1.1.2.2.1.** El valor por defecto de MAX_NUM_KEYWORDS_TOPIC_TEXTO es de 30.

**RF-3.1.1.2.2.2.** MAX_NUM_KEYWORDS_TOPIC_TEXTO debe poder ser configurado por el administrador del sistema.

**RF-3.1.1.2.2.2.1.** El sistema verificará que el valor de MAX_NUM_KEYWORDS_TOPIC_TEXTO sea mayor de 0.

**RF-3.1.1.3.** Por cada keyword, el sistema mostrará su probabilidad dentro del topic.

**RF-3.2.** Los topics se podrán visualizar en formato nube de palabras.

**RF-3.2.1.** Cada nube de palabras relativa a un topic estará formada por las NUM_KEYWORDS_TOPIC_NUBE keywords más relevantes de dicho topic.

**RF-3.2.1.1.** El valor por defecto de NUM_KEYWORDS_TOPIC_NUBE es de 10.

**RF-3.2.1.2.** NUM_KEYWORDS_TOPIC_NUBE debe poder ser configurado por el usuario.

**RF-3.2.1.2.1.** El sistema verificará que el valor de NUM_KEYWORDS_TOPIC_NUBE sea mayor de 0.

**RF-3.2.1.2.2.** El sistema verificará que el valor de NUM_KEYWORDS_TOPIC_NUBE sea menor o igual que MAX_NUM_KEYWORDS_TOPIC_NUBE.

**RF-3.2.1.2.2.1.** El valor por defecto de MAX_NUM_KEYWORDS_TOPIC_NUBE es de 100.

**RF-3.2.1.2.2.2.** MAX_NUM_KEYWORDS_TOPIC_NUBE debe poder ser configurado por el administrador del sistema.

**RF-3.2.1.2.2.2.1.** El sistema verificará que el valor de MAX_NUM_KEYWORDS_TOPIC_NUBE sea mayor de 0.

**RF-3.2.2.** Cada nube de palabras relativa a un topic establecerá el tamaño de las keywords en función de su probabilidad dentro del topic.

**RF-3.2.2.1.** Las keywords más importantes tendrán un tamaño superior a las menos importantes.

**RF-3.3.** En ambos casos (RF-3.1 y RF-3.2), también se mostrará el número de cada topic.

**RF-4.** El sistema permitirá al usuario visualizar los NUM_DOCUMENTOS_TOPIC documentos más representativos de un topic.

**RF-4.1.** El valor por defecto de NUM_DOCUMENTOS_TOPIC es de 6.

**RF-4.2.** NUM_DOCUMENTOS_TOPIC debe poder ser configurado por el usuario.

**RF-4.2.1.** El sistema verificará que el valor de NUM_DOCUMENTOS_TOPIC sea mayor de 0.

**RF-4.2.2.** El sistema verificará que el valor de NUM_DOCUMENTOS_TOPIC sea menor o igual que MAX_NUM_DOCUMENTOS_TOPIC.

**RF-4.2.2.1.** El valor por defecto de MAX_NUM_DOCUMENTOS_TOPIC es de 30.

**RF-4.2.2.2.** MAX_NUM_DOCUMENTOS_TOPIC debe poder ser configurado por el administrador del sistema.

**RF-4.2.2.2.1.** El sistema verificará que el valor de MAX_NUM_DOCUMENTOS_TOPIC sea mayor de 0.

**RF-4.3.** El sistema permitirá al usuario seleccionar el topic del cuál obtener los documentos más representativos.

**RF-4.4.** Por cada documento se mostrará:

**RF-4.4.1.** Un resumen extractivo de su contenido.

**RF-4.4.1.1.** El resumen constará de NUM_FRASES_RESUMEN_ADMIN frases, extraídas del texto de entrada.

**RF-4.4.1.1.1.** El valor por defecto de NUM_FRASES_RESUMEN_ADMIN es de 2.

**RF-4.4.1.1.2.** NUM_FRASES_RESUMEN_ADMIN debe poder ser configurado por el administrador del sistema.

**RF-4.4.1.1.2.1.** El sistema verificará que el valor de NUM_FRASES_RESUMEN_ADMIN sea mayor de 0.

**RF-4.4.1.1.3.** Si NUM_FRASES_RESUMEN_ADMIN es mayor que el número de frases del documento, el resumen constará de todas las frases del documento.

**RF-4.4.2.** El contenido completo del documento.

**RF-4.4.3.** La probabilidad de que el documento esté relacionado con el topic.

**RF-5.** El sistema mostrará al usuario los topics más relacionados con un determinado texto.

**RF-5.1.** El sistema permitirá al usuario introducir dicho texto.

**RF-5.2.** El sistema permitirá al usuario establecer el número máximo de topics que serán devueltos.

**RF-5.2.1.** El sistema comprobará que el número indicado por el usuario sea mayor de 0.

**RF-5.2.2.** El sistema comprobará que el número indicado por el usuario sea menor o igual que el número de topics del modelo de extracción de topics.

**RF-5.2.3.** El número de topics devueltos por el sistema será el mínimo entre el número indicado por el usuario y el número de topics devueltos por el modelo de extracción de topics.

**RF-5.3.** El sistema mostrará, por cada topic que devuelva:

**RF-5.3.1.** El número del topic.

**RF-5.3.2.** La probabilidad de que el texto esté relacionado con ese topic.

**RF-5.4.** Los topics devueltos estarán ordenados por la probabilidad indicada en RF-5.3.

**RF-5.5.** El sistema mostrará también una visualización gráfica comparando la probabilidad de que los topics devueltos estén relacionados con el texto.

**RF-6.** El sistema mostrará al usuario, dado un texto de entrada, una serie de documentos similares a dicho texto.

**RF-6.1.** El sistema permitirá al usuario introducir dicho texto.

**RF-6.2.** Los documentos similares serán documentos pertenecientes al dataset con el que se ha entrenado el modelo de obtención de topics.

**RF-6.3.** La similitud de los documentos devueltos con el texto de entrada será calculada teniendo en cuenta:

**RF-6.3.1.** Los topics más relacionados con el texto de entrada.

**RF-6.3.1.1.** Por cada topic se tendrá en cuenta también la probabilidad de que el texto de entrada esté relacionado con dicho topic.

**RF-6.3.2.** Los documentos más representativos de los topics más relacionados con el texto de entrada.

**RF-6.3.2.1.** Por cada documento se tendrá en cuenta también la probabilidad de que esté relacionado con su topic más representativo.

**RF-6.4.** El sistema permitirá al usuario visualizar los NUM_DOCUMENTOS_SIMILARES documentos más similares.

**RF-6.4.1.** El valor por defecto de NUM_DOCUMENTOS_SIMILARES es de 6.

**RF-6.4.2.** NUM_DOCUMENTOS_SIMILARES debe poder ser configurado por el usuario.

**RF-6.4.2.1.** El sistema verificará que el valor de NUM_DOCUMENTOS_SIMILARES sea mayor de 0.

**RF-6.4.2.2.** El sistema verificará que el valor de NUM_DOCUMENTOS_SIMILARES sea menor o igual que MAX_NUM_DOCUMENTOS_SIMILARES.

**RF-6.4.2.2.1.** El valor por defecto de MAX_NUM_DOCUMENTOS_SIMILARES es de 20.

**RF-6.4.2.2.2.** MAX_NUM_DOCUMENTOS_SIMILARES debe poder ser configurado por el administrador del sistema.

**RF-6.4.2.2.2.1.** El sistema verificará que el valor de MAX_NUM_DOCUMENTOS_SIMILARES sea mayor de 0.

**RF-6.5.** Por cada documento se mostrará:

**RF-6.5.1.** Lo indicado en RF-4.4.1.

**RF-6.5.2.** Lo indicado en RF-4.4.2.

**RF-6.5.3.** La probabilidad de que dicho documento sea similar al texto de entrada.

**RF-6.5.4.** El número del topic con el que más se relaciona el documento.

**RF-7.** El sistema mostrará al usuario, dado un texto de entrada, un resumen extractivo del mismo.

**RF-7.1.** El sistema permitirá al usuario introducir dicho texto.

**RF-7.2.** El resumen constará de NUM_FRASES_RESUMEN_USUARIO frases, extraídas del texto de entrada.

**RF-7.2.1.** El valor por defecto de NUM_FRASES_RESUMEN_USUARIO es de 4.

**RF-7.2.2.** NUM_FRASES_RESUMEN_USUARIO debe poder ser configurado por el usuario.

**RF-7.2.2.1.** El sistema verificará que el valor de NUM_FRASES_RESUMEN_USUARIO sea mayor de 0.

**RF-7.2.2.2.** Si NUM_FRASES_RESUMEN_USUARIO es mayor que el número de frases del texto de entrada, el resumen constará de todas las frases del texto.

**RF-8.** El sistema permitirá al administrador configurar todos los parámetros del sistema mencionados previamente mediante una interfaz web.

**RF-8.1.** El sistema le mostrará el valor actual de cada parámetro.

**RF-8.2.** El sistema le permitirá modificar el valor de cada parámetro.

#### *5.2.1.3 Requisitos no funcionales*

##### 5.2.1.3.1 Requisitos tecnológicos

**RNF-TEC-1.** El backend será implementado utilizando el framework Flask.

**RNF-TEC-2.** El frontend será implementado utilizando el framework Angular.

**RNF-TEC-3.** El backend será encapsulado dentro de un contenedor Docker para su despliegue.

**RNF-TEC-4.** El frontend será encapsulado dentro de un contenedor Docker para su despliegue.

##### 5.2.1.3.2 Requisitos de adaptabilidad

**RNF-ADA-1.** El sistema deberá poder ser utilizado correctamente en tamaños de pantalla cuyo ancho sea mayor o igual a 320 píxeles.

##### 5.2.1.3.3 Requisitos de seguridad

**RNF-SEG-1.** El administrador del sistema deberá identificarse mediante nombre de usuario y contraseña.

**RNF-SEG-2.** El sistema almacenará la contraseña del administrador de forma cifrada.

**RNF-SEG-2.1.** La forma de cifrar la contraseña será mediante una función hash.

### 5.2.2 Identificación de Actores del Sistema

- **Usuario:** Persona que interacciona con la interfaz web pública de la aplicación.
- **Administrador:** Persona que interacciona con el sistema para gestionar sus parámetros.

### 5.2.3 Especificación de Casos de Uso

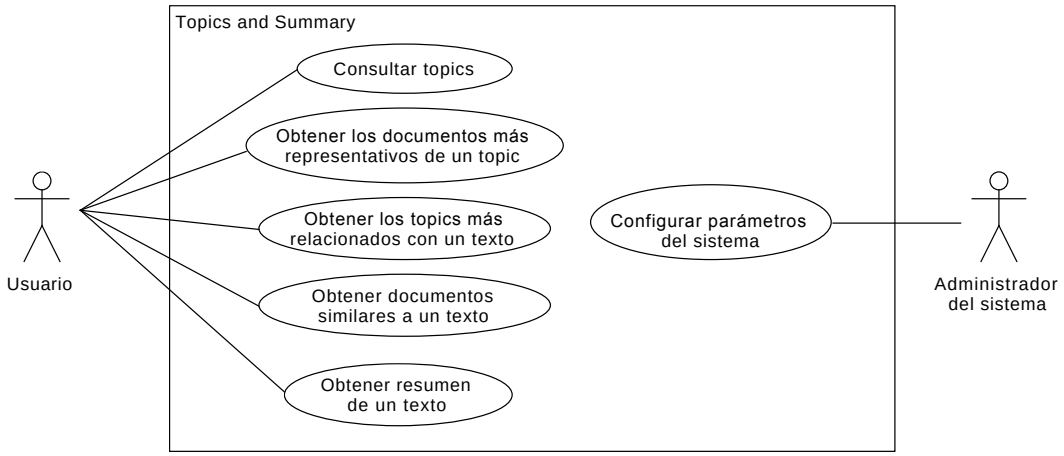



*Figura 5.1: Diagrama de contexto*

| **Nombre del Caso de Uso** |
|---|
| Consultar topics |
| **Descripción** |
| El usuario visualizará los topics que han sido generados por el modelo. El sistema le permitirá seleccionar si quiere visualizar los topics en formato textual o en formato "nube de palabras":<br>• Si selecciona el formato textual, el sistema le mostrará una lista con los topics, y por cada topic le mostrará las keywords más importantes, de forma textual, junto a la importancia de cada keyword dentro del topic.<br>• Si selecciona el formato "nube de palabras", el sistema le mostrará imágenes con los topics, donde cada topic estará formado por sus keywords más importantes. La importancia de las keywords dentro de un topic vendrá dada por el tamaño de las mismas (keywords más grandes son más importantes dentro del topic).<br>En ambos casos, el usuario podrá seleccionar el número de keywords por topic. |

| **Nombre del Caso de Uso** |
|---|
| Obtener los documentos más representativos de un topic |
| **Descripción** |
| El usuario seleccionará un topic para obtener los documentos más representativos de dicho topic. El usuario podrá indicar cuantos documentos quiere que se devuelvan. El sistema le devolverá los documentos más representativos del topic, mostrando por cada uno:<br>• Un pequeño resumen de su contenido |

| |
|---|
| • El contenido completo del documento<br>• La probabilidad de que el documento esté relacionado con el topic |

| Nombre del Caso de Uso |
|---|
| Obtener los topics más relacionados con un texto |
| **Descripción** |
| El usuario introducirá un texto y seleccionará la opción para mostrar los topics relacionados con dicho texto. El sistema le devolverá los topics más relacionados, y por cada uno de ellos le mostrará la probabilidad de que el texto esté relacionado con el topic.<br>El sistema le mostrará también una visualización gráfica, en la que se compararán las probabilidades de que los topics estén relacionados con el texto introducido por el usuario. |

| **Nombre del Caso de Uso** |
|---|
| Obtener documentos similares a un texto |
| **Descripción** |
| El usuario introducirá un texto y seleccionará la opción para mostrar los documentos similares a dicho texto. El sistema le devolverá, de los documentos con los que se entrenó el modelo, aquellos que sean más similares al texto de entrada. El usuario podrá indicar cuantos documentos quiere que se devuelvan. El sistema le devolverá un conjunto de documentos, mostrando por cada uno:<br>• Un pequeño resumen de su contenido<br>• El contenido completo del documento<br>• La probabilidad de que el documento y el texto introducido por el usuario sean similares |

| **Nombre del Caso de Uso** |
|---|
| Obtener resumen de un texto |
| **Descripción** |
| El usuario introducirá un texto y seleccionará la opción para resumir el texto. El sistema le devolverá un resumen del mismo, formado por sus frases más relevantes. El usuario podrá indicar cuantas frases del texto de entrada quiere que se devuelvan. |

| **Nombre del Caso de Uso** |
|---|
| Configurar parámetros del sistema |
| **Descripción** |
| El sistema le permitirá al administrador configurar los parámetros del sistema, es decir, obtener sus valores actuales y modificarlos. Estos parámetros se refieren a los valores máximos y mínimos que limitan los valores que puede introducir el usuario en el resto de casos de uso. |

## 5.3 Identificación de los Subsistemas en la Fase de Análisis

### 5.3.1 Descripción de los Subsistemas

Los subsistemas que se han identificado son los siguientes:

- **Topics and Summary:** Conjunto de paquetes y clases que contiene todos los elementos necesarios para la generación, análisis, comparación, selección, almacenamiento y utilización de modelos de extracción de topics y generación de resúmenes de textos.
- **Web backend:** Backend de la parte web del sistema. Este subsistema, que se implementará utilizando el framework *Flask*, se comunica tanto con el subsistema *Topics and Summary* como con el subsistema *Web Frontend*, y es el punto de unión entre ambos, ya que permite transferir información desde el frontend hacia los modelos, y viceversa.
- **Web frontend:** Frontend de la parte web del sistema. Este subsistema, que se implementará utilizando el framework *Angular*, proporciona una interfaz web, que permitirá al usuario interaccionar con el sistema.

### 5.3.2 Descripción de los Interfaces entre Subsistemas

El subsistema *Topics and Summary* ofrece una API, que será utilizada por el subsistema *Web backend* para acceder a los modelos generados por el primero. Ambos subsistemas se comunicarán localmente, dado que estarán desplegados en la misma máquina.

Por otro lado, el subsistema *Web backend* y el subsistema *Web frontend* se comunicarán a través de un servicio web REST. Por este motivo, ambos subsistemas podrán ser desplegados en máquinas distintas, dado que se podrán comunicar a través de la red, mediante el protocolo HTTP.

# 5.4 Diagrama de Clases Preliminar del Análisis

## 5.4.1 Diagrama de Clases

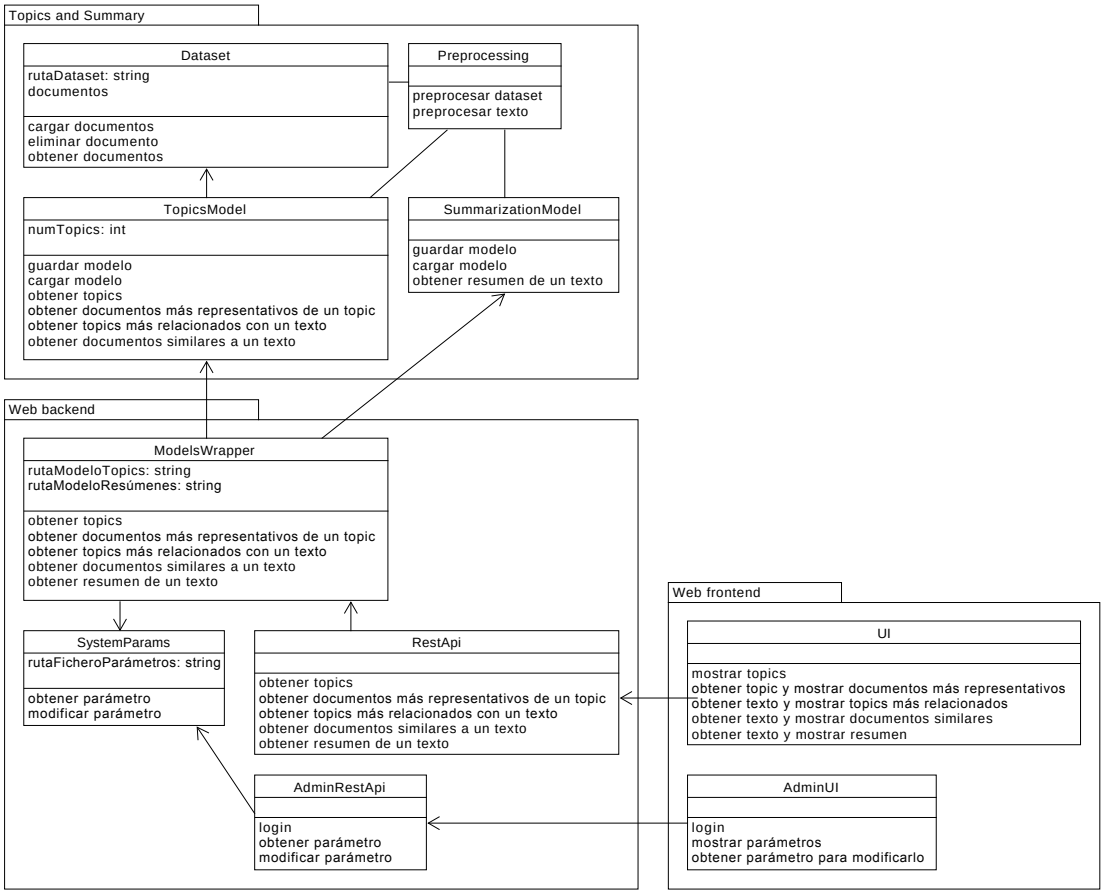



*Figura 5.2: Diagrama de clases de análisis*

## 5.4.2 Descripción de las Clases

### *5.4.2.1 Topics and Summary*

| **Nombre de la Clase** |
|---|
| Dataset |
| Descripción |
| Clase que encapsula los documentos del dataset utilizado para entrenar los modelos |
| Responsabilidades |
| • Cargar en memoria los documentos del dataset<br>• Permitir el acceso a los documentos cargados en memoria |
| Atributos Propuestos |
| **rutaDataset**: Almacena la ruta donde se encuentra almacenado el dataset en disco<br>**documentos**: Almacena los documentos del dataset |
| Métodos Propuestos |

| |
|---|
| **cargar documentos**: Carga en memoria los documentos del dataset, que están almacenados en disco<br>**eliminar documento**: Elimina del atributo documentos el documento especificado<br>**obtener documentos:** Devuelve los documentos cargados en memoria |

| **Nombre de la Clase** |
|---|
| Preprocessing |
| Descripción |
| Clase que encapsula las operaciones de preprocesado |
| Responsabilidades |
| • Preprocesar los documentos del dataset<br>• Preprocesar texto |
| Atributos Propuestos |
| |
| Métodos Propuestos |
| **preprocesar dataset**: Permite preprocesar los documentos del dataset<br>**preprocesar texto**: Permite preprocesar un texto |

| **Nombre de la Clase** |
|---|
| TopicsModel |
| Descripción |
| Clase que encapsula la generación, el almacenamiento y la utilización de modelos de extracción de topics |
| Responsabilidades |
| • Generar modelos de extracción de topics, utilizando los documentos del dataset, después de haber sido preprocesados<br>• Almacenar en disco los modelos generados, para su posterior utilización<br>• Cargar en memoria modelos almacenados en disco<br>• Devolver los topics generados por el modelo<br>• Devolver los documentos más representativos de un determinado topic<br>• Devolver los topics más relacionados con un determinado texto<br>• Devolver los documentos más similares a un determinado texto |
| Atributos Propuestos |
| **numTopics**: Almacena el número de topics del modelo generado |
| Métodos Propuestos |
| **guardar modelo:** Permite guardar el modelo en disco<br>**cargar modelo:** Permite cargar el modelo de disco<br>**obtener topics:** Permite obtener los topics generados por el modelo<br>**obtener documentos más representativos de un topic:** Permite obtener los documentos más representativos de un determinado topic<br>**obtener topics más relacionados con un texto:** Permite obtener los topics más relacionados con un determinado texto<br>**obtener documentos similares a un texto:** Permite obtener los documentos más similares a un determinado texto |

| **Nombre de la Clase** |
|---|
| SummarizationModel |
| Descripción |
| Clase que encapsula la generación, el almacenamiento y la utilización de modelos para generar resúmenes de textos |
| Responsabilidades |
| • Generar modelos que permitan realizar resúmenes de textos<br>• Almacenar en disco los modelos generados, para su posterior utilización<br>• Cargar en memoria modelos almacenados en disco<br>• Devolver el resumen de un determinado texto |
| Atributos Propuestos |
| |
| Métodos Propuestos |
| **guardar modelo:** Permite guardar el modelo en disco<br>**cargar modelo:** Permite cargar el modelo de disco<br>**obtener resumen de un texto** Permite obtener el resumen de un determinado texto |

#### 5.4.2.2 Web backend

| **Nombre de la Clase** |
|---|
| SystemParams |
| Descripción |
| Clase que encapsula el acceso y la modificación de los parámetros del sistema. |
| Responsabilidades |
| • Permitir el acceso a los parámetros del sistema, almacenados en un fichero<br>• Permitir la modificación de los parámetros del sistema, almacenados en un fichero |
| Atributos Propuestos |
| **rutaFicheroParámetros**: Almacena la ruta donde está guardado en disco el fichero con los parámetros del sistema |
| Métodos Propuestos |
| **obtener parámetro**: Dado el nombre de un parámetro, devuelve su valor.<br>**modificar parámetro**: Dado el nombre de un parámetro y su nuevo valor, modifica el valor de dicho parámetro. |

| **Nombre de la Clase** |
|---|
| ModelsWrapper |
| Descripción |
| Clase que encapsula las operaciones del modelo de extracción de topics y del modelo para generar resúmenes de textos. Por tanto, se comunica con el subsistema *Topics and Summary* (en particular, con las clases *TopicsModel* y *SummarizationModel*). |
| Responsabilidades |
| • Cargar de disco el modelo de extracción de topics y el modelo de generación de resúmenes<br>• Ofrecer una interfaz simplificada para la utilización de ambos modelos<br>• Comunicarse con la clase *SystemParams* para acceder a los parámetros del sistema<br>• Realizar validaciones de los valores de entrada a los métodos de la clase, utilizando para ello los parámetros del sistema |
| Atributos Propuestos |

| |
|---|
| **rutaModeloTopics**: Almacena la ruta donde está guardado en disco el modelo de extracción de topics<br>**rutaModeloResúmenes**: Almacena la ruta donde está guardado en disco el modelo de generación de resúmenes |
| Métodos Propuestos |
| **obtener topics**: Encapsula el método *obtener topics* del objeto *TopicsModel*.<br>**obtener documentos más representativos de un topic**: Encapsula el método *obtener documentos más representativos de un topic* del objeto *TopicsModel*.<br>**obtener topics más relacionados con un texto**: Encapsula el método *obtener topics más relacionados con un texto* del objeto *TopicsModel*.<br>**obtener documentos similares a un texto**: Encapsula el método *obtener documentos similares a un texto* del objeto *TopicsModel*.<br>**obtener resumen de un texto**: Encapsula el método *obtener resumen de un texto* del objeto *SummarizationModel*. |

| **Nombre de la Clase** |
|---|
| RestApi |
| Descripción |
| Clase que ofrece un servicio web REST para comunicarse con los modelos de extracción de topics y de generación de resúmenes a través de la clase *ModelsWrapper*. |
| Responsabilidades |
| • Ofrecer una API REST para permitir al usuario acceder a información de los modelos y permitirle interaccionar con sus funcionalidades principales<br>• Comunicarse con la clase *ModelsWrapper* para obtener la información de los modelos y acceder a sus funcionalidades principales |
| Atributos Propuestos |
| |
| Métodos Propuestos |
| **obtener topics:** Llama al método *obtener topics* de la clase *ModelsWrapper* y devuelve su resultado mediante el servicio web.<br>**obtener documentos más representativos de un topic:** Llama al método *obtener documentos más representativos de un topic* de la clase *ModelsWrapper* y devuelve su resultado mediante el servicio web.<br>**obtener topics más relacionados con un texto:** Llama al método *obtener topics más relacionados con un texto* de la clase *ModelsWrapper* y devuelve su resultado mediante el servicio web.<br>**obtener documentos similares a un texto:** Llama al método *obtener documentos similares a un texto* de la clase *ModelsWrapper* y devuelve su resultado mediante el servicio web.<br>**obtener resumen de un texto:** Llama al método *obtener resumen de un texto* de la clase *ModelsWrapper* y devuelve su resultado mediante el servicio web. |

| **Nombre de la Clase** |
|---|
| AdminRestApi |
| Descripción |
| Clase que ofrece un servicio web REST que permite al administrador del sistema obtener y modificar los parámetros del sistema. |
| Responsabilidades |
| • Ofrecer una API REST para permitir al administrador del sistema obtener y modificar los valores de los parámetros del sistema |

| |
|---|
| • Añadir un concepto de login, que solicite las credenciales del administrador para permitir el acceso al resto de métodos de la API<br>• Comunicarse con la clase *SystemParams* para acceder a los parámetros del sistema, que están almacenados en un fichero |
| Atributos Propuestos |
| |
| Métodos Propuestos |
| **login:** Permite introducir un nombre de usuario y una contraseña, y si coinciden con los del administrador, devuelve un token de seguridad, que será necesario para realizar el resto de las peticiones de esta API REST.<br>**obtener parámetro:** Llama al método *obtener parámetro* de la clase *SystemParams* y devuelve su resultado mediante el servicio web.<br>**modificar parámetro:** Llama al método *modificar parámetro* de la clase *SystemParams* con los valores especificados en la llamada al servicio web. |

#### 5.4.2.3 Web frontend

| **Nombre de la Clase** |
|---|
| UI |
| Descripción |
| Clase que encapsula la funcionalidad necesaria para recibir las interacciones del usuario, comunicarse con la API REST definida en la clase *RestApi* del subsistema *Web backend*, y mostrar los resultados devueltos por dicha API a través de la interfaz web. |
| Responsabilidades |
| • Ofrecer una interfaz web que permita al usuario acceder a las funcionalidades ofrecidas por la clase *RestApi* del subsistema *Web backend* |
| Atributos Propuestos |
| |
| Métodos Propuestos |
| **mostrar topics:** Se comunica con la URI ofrecida por el método *obtener topics* de la clase *RestApi* para obtener los topics, y los muestra por la interfaz web.<br>**obtener topic y mostrar documentos más representativos:** Permite al usuario seleccionar un topic mediante la interfaz web, se comunica con la URI ofrecida por el método *obtener documentos más representativos de un topic* de la clase *RestApi*, enviándole dicho topic, y muestra los documentos devueltos por el servicio REST mediante la interfaz web.<br>**obtener texto y mostrar topics más relacionados:** Permite al usuario introducir un texto mediante la interfaz web, se comunica con la URI ofrecida por el método *obtener topics más relacionados con un texto* de la clase *RestApi*, enviándole dicho texto, y muestra los topics devueltos por el servicio REST mediante la interfaz web.<br>**obtener texto y mostrar documentos similares:** Permite al usuario introducir un texto mediante la interfaz web, se comunica con la URI ofrecida por el método *obtener documentos similares a un texto* de la clase *RestApi*, enviándole dicho texto, y muestra los documentos devueltos por el servicio REST mediante la interfaz web.<br>**obtener texto y mostrar resumen:** Permite al usuario introducir un texto mediante la interfaz web, se comunica con la URI ofrecida por el método *obtener resumen de un texto* de la clase *RestApi*, enviándole dicho texto, y muestra el resumen devuelto por el servicio REST mediante la interfaz web. |

| **Nombre de la Clase** |
|---|
| AdminUI |
| Descripción |
| Clase que encapsula la funcionalidad necesaria para permitir al administrador del sistema iniciar sesión y acceder a los parámetros del sistema, mediante una interfaz web. Para ello, se comunica con la API REST definida en la clase *AdminRestApi* del subsistema *Web backend*. |
| Responsabilidades |
| • Ofrecer una interfaz web que permita al administrador iniciar sesión y acceder a los parámetros del sistema |
| Atributos Propuestos |
| |
| Métodos Propuestos |
| **login:** Permite al administrador introducir su nombre de usuario y su contraseña mediante la interfaz web, se comunica con la URI ofrecida por el método *login* de la clase *AdminRestApi*, enviándole dichas credenciales, y si coinciden con las del administrador, le permite acceder a los parámetros del sistema.<br>**mostrar parámetros:** Permite al administrador visualizar los parámetros del sistema mediante la interfaz web. Para ello, se comunica con la URI ofrecida por el método *obtener parámetro* de la clase *AdminRestApi*, enviándole el token de seguridad obtenido al realizar el login.<br>**obtener parámetro para modificarlo:** Permite al administrador introducir un nuevo valor para un parámetro del sistema mediante la interfaz web, y se comunica con la URI ofrecida por el método *modificar parámetro* de la clase *AdminRestApi*, enviándole el token de seguridad obtenido al realizar el login y el nuevo valor del parámetro. |

## 5.5 Análisis de Interfaces de Usuario

### 5.5.1 Descripción de la Interfaz

#### *5.5.1.1 Interfaz principal*

A continuación, se pasa a describir la interfaz principal, la cual ofrece la funcionalidad indicada en los casos de uso relativos al actor "usuario". La interfaz se divide en dos zonas, como se puede ver en la Figura 5.3.

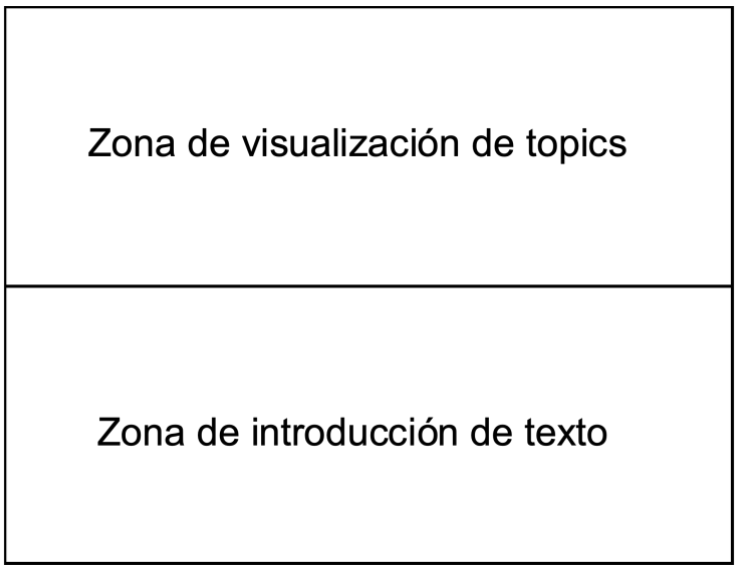



*Figura 5.3: Boceto de las zonas de la interfaz de usuario*

En la zona superior, se muestran los topics y se pueden visualizar los documentos más representativos de un topic.

En la zona inferior, se puede introducir un texto para mostrar: los topics relacionados, documentos similares o un resumen del texto.

A continuación, se muestran 2 bocetos detallados de la interfaz principal:

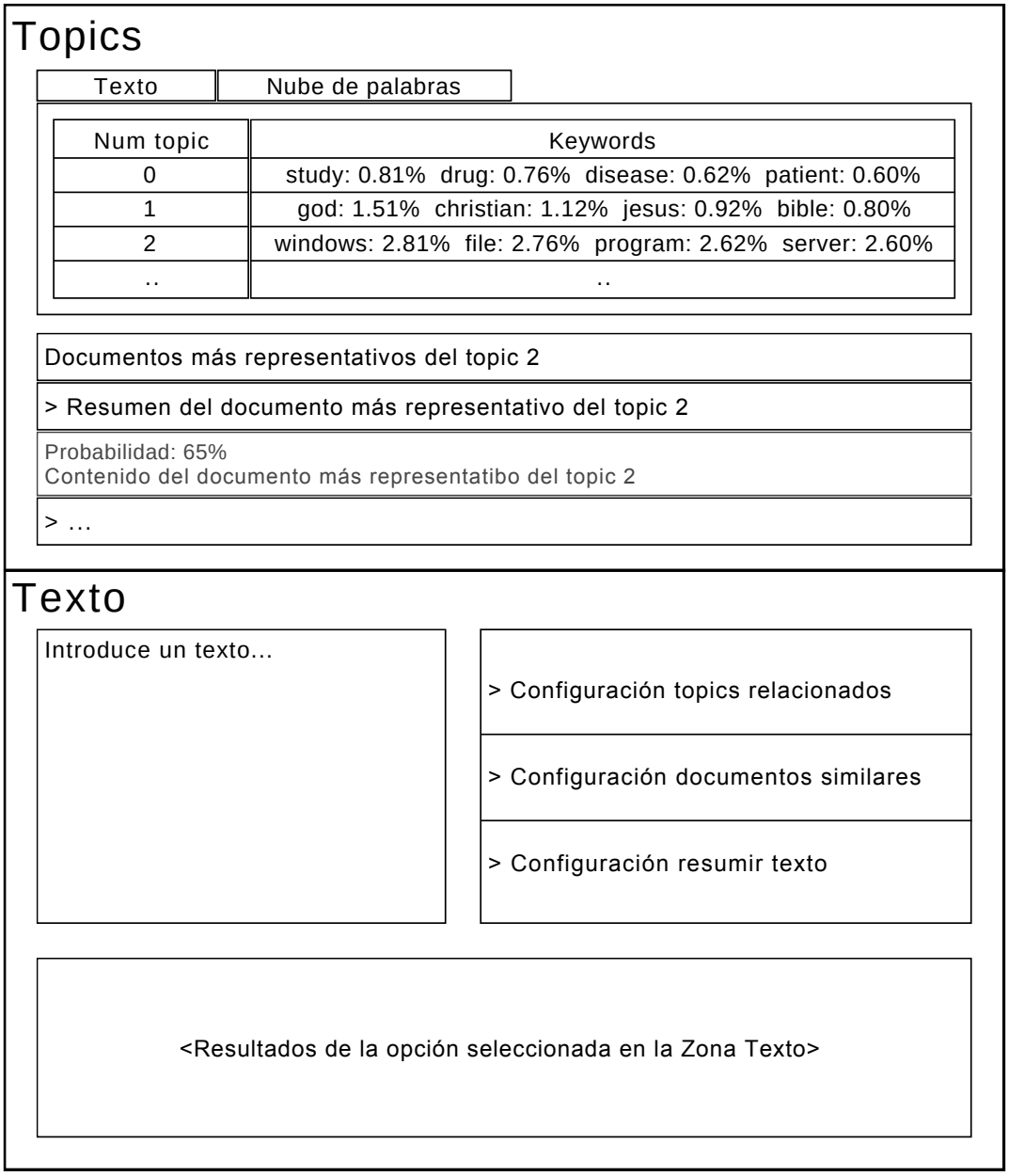



*Figura 5.4: Boceto detallado de la interfaz de usuario (mostrando la pestaña Topics-Texto)*

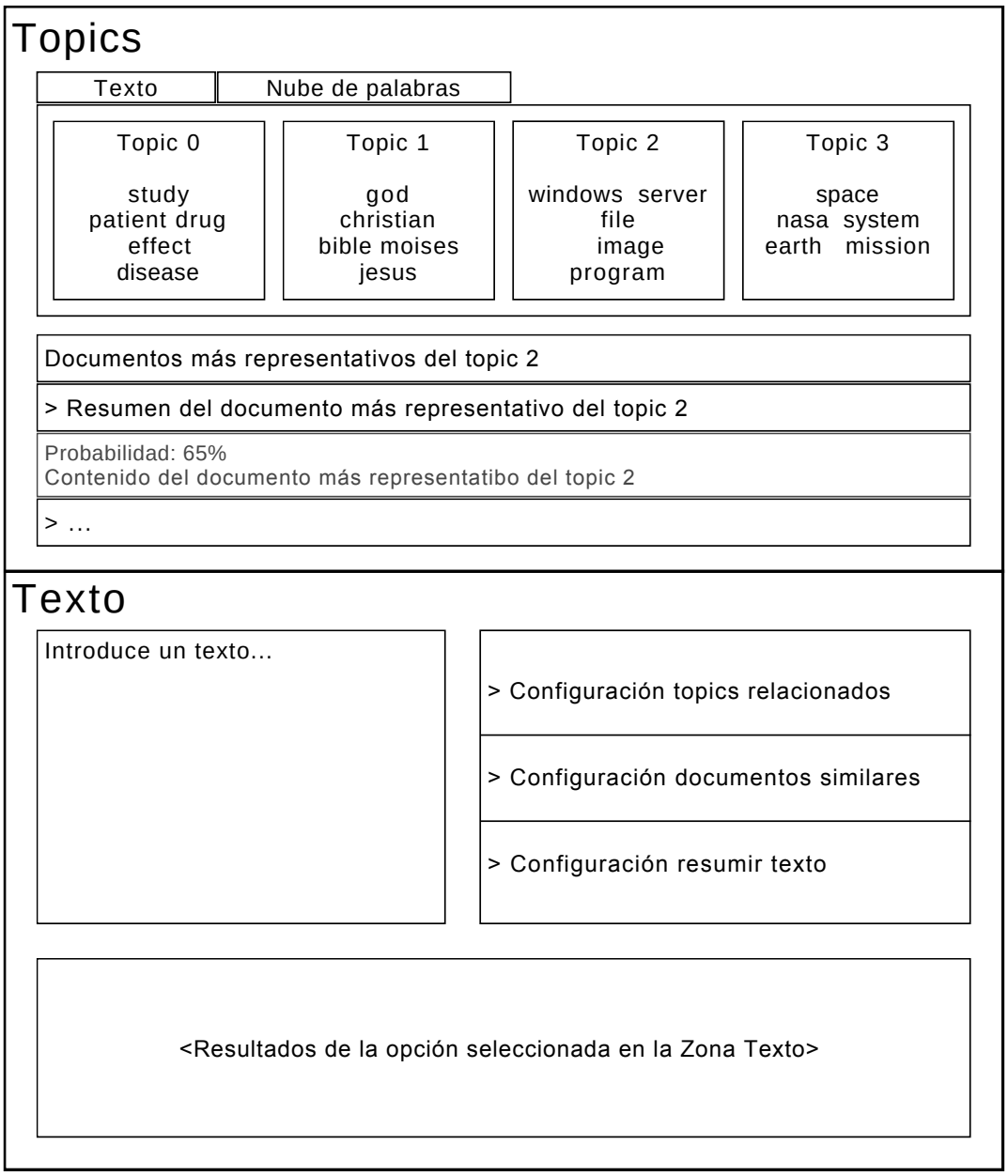



*Figura 5.5: Boceto detallado de la interfaz de usuario (mostrando la pestaña Topics-Nube de palabras)*

### 5.5.2 Descripción del Comportamiento de la Interfaz

#### *5.5.2.1 Interfaz principal*

La interfaz mantiene la división en zonas indicada en la Figura 5.3 en todo momento. No hay cambios de pantalla como tal, sino que se trata de una pantalla única, en la que se va mostrando y ocultando información de forma dinámica.

En la zona superior, se muestra un panel con 2 pestañas, que permite cambiar entre las 2 formas disponibles de visualizar los topics: texto y nube de palabras. También se dispone de un icono de configuración, en la parte superior derecha, el cual despliega un diálogo que permite configurar los parámetros de la parte superior de la interfaz. Al hacer click en el icono de un documento presente en una de las filas de los topics, dentro de la visualización en formato texto; o al hacer click en el botón "Documentos más representativos" situado debajo de un topic, dentro de la visualización en formato nube de palabras; aparecen debajo, de forma dinámica, los documentos más representativos de dicho topic. En la visualización en formato nube de palabras, arriba a la izquierda, se incluye un icono de ayuda, que indica al usuario que puede ampliar las imágenes haciendo click en ellas.

En la zona inferior, se dispone de un cuadro de texto, donde el usuario deberá introducir el texto a utilizar. A la derecha de dicho cuadro de texto se dispone de un panel con 3 desplegables. Cada uno de ellos permite configurar y obtener los resultados de las 3 funcionalidades restantes: topics relacionados, documentos similares y resumen del texto. Los resultados de estas 3 funciones aparecerán, nuevamente, de forma dinámica, debajo del cuadro de texto y del panel de configuración.

Toda la información que aparece dinámicamente debajo de cada una de las 2 zonas puede ser ocultada haciendo click en un icono con forma de x.

### 5.5.3 Diagrama de Navegabilidad

#### *5.5.3.1 Interfaz principal*

Como se ha mencionado en el apartado anterior, se dispone de una única pantalla, en la que la nueva información se muestra dinámicamente. Por tanto, no existe una navegación preestablecida como tal. La información que se muestra en un momento dado en pantalla puede ser obtenida de diversas formas.

## 5.6 Especificación del Plan de Pruebas

Las pruebas que se realizarán son las siguientes:

- **Pruebas unitarias automatizadas:** Se realizarán pruebas unitarias automatizadas de, al menos, los métodos descritos en el de todas las "clases del análisis" del subsistema *Topics and Summary,* así como de la "clase del análisis" *SystemParams* del subsistema *Web backend*. Estas pruebas unitarias se realizarán utilizando la librería *unittest*, de la biblioteca estándar de Python.
- **Análisis estático de código:** Se analizará el código fuente de los subsistemas *Topics and Summary* y *Web backend*, utilizando la herramienta *SonarQube*.
- **Pruebas funcionales no automatizadas:** Además de las pruebas automatizadas, todas las partes de la aplicación no cubiertas por éstas últimas serán probadas de forma manual, verificando así su correcto funcionamiento.
- **Pruebas de usabilidad:** Se realizarán pruebas de usabilidad de las funcionalidades principales del sistema, siendo éstas las funciones de la clase *UI* del subsistema *Web frontend.* Para ello, se solicitará a 3 participantes, con distintos perfiles y edades, que interaccionen con el sistema, una vez esté en su fase final de desarrollo. Los resultados de estas pruebas se recogerán utilizando 2 cuestionarios, que están descritos en la sección Cuestionarios.
- **Pruebas de adaptabilidad:** Se realizarán pruebas para comprobar si el subsistema *Web frontend* es adaptable a distintos tamaños de pantalla, prestando especial atención a las pantallas de dispositivos móviles.

# Capítulo 6. Diseño del Sistema

## 6.1 Arquitectura del Sistema

A continuación, se muestran los diagramas de componentes y de despliegue.

### 6.1.1 Diagramas de Componentes

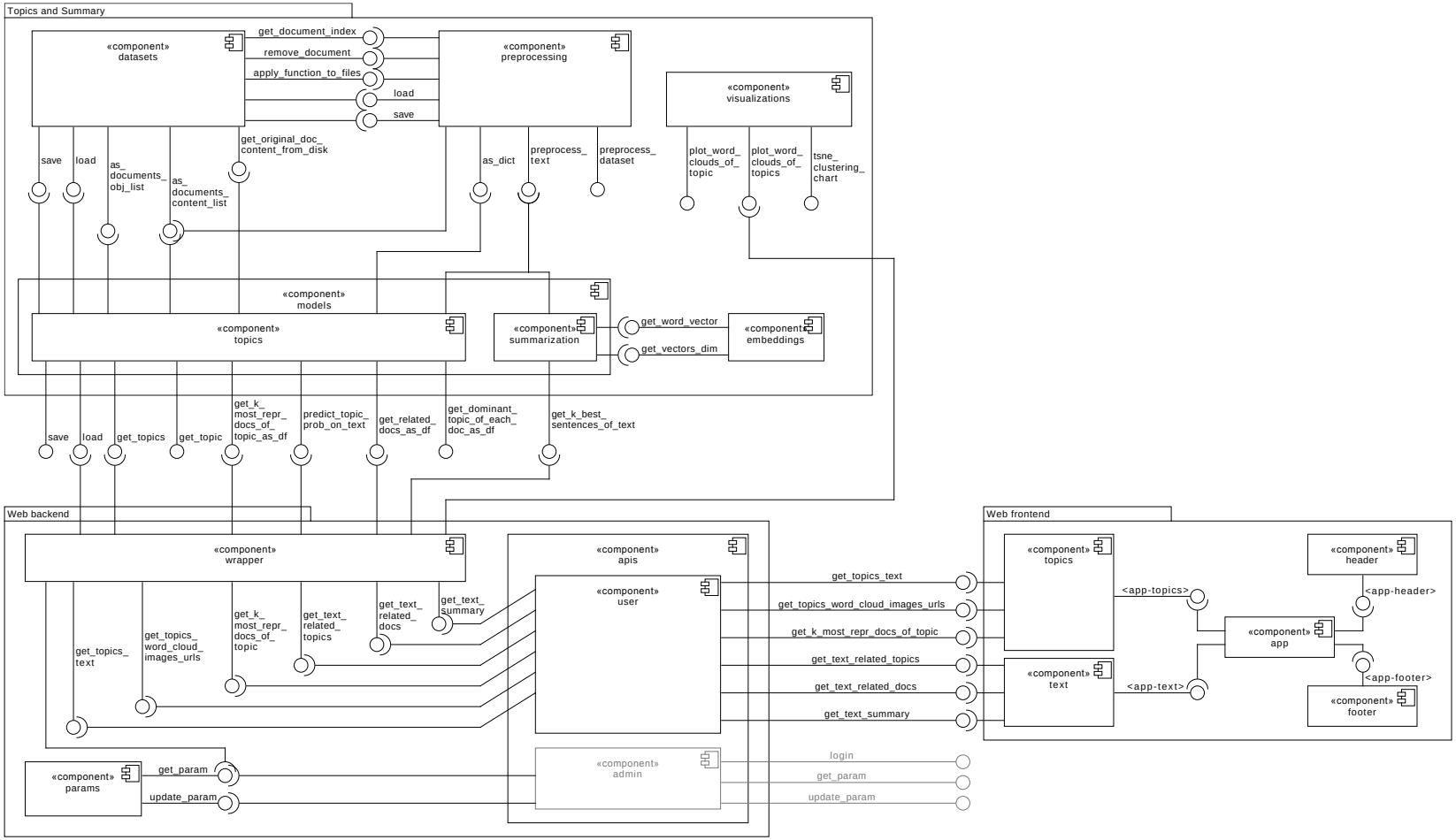



*Figura 6.1: Diagrama de Componentes*

## 6.1.2 Diagramas de Despliegue

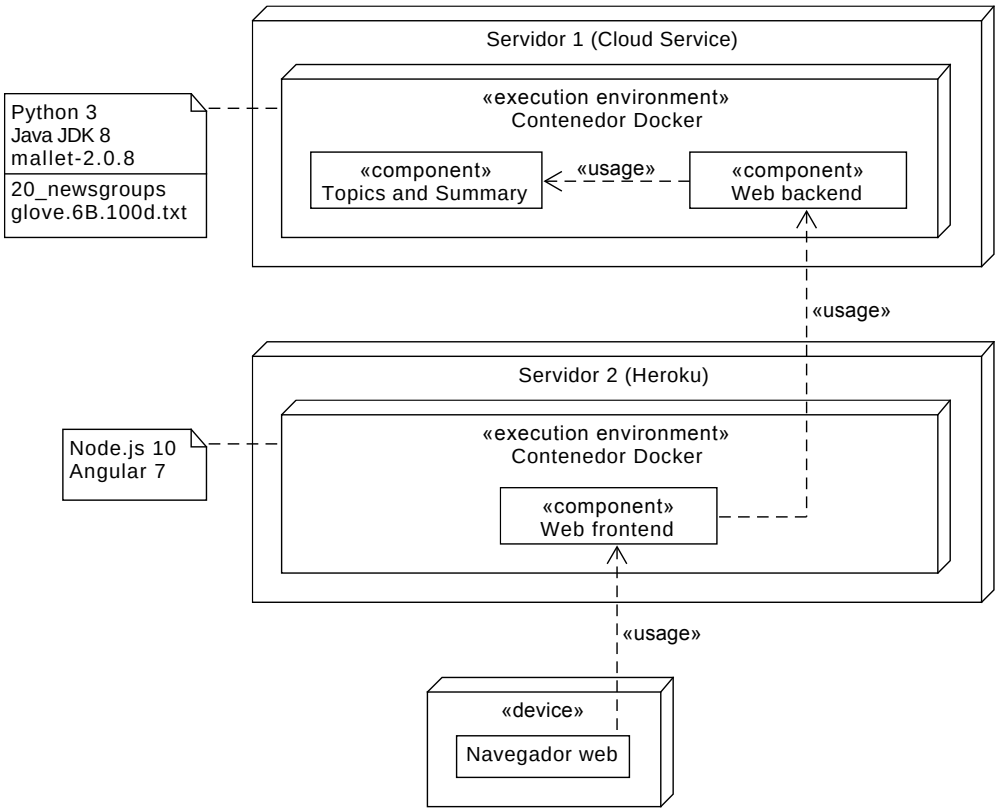



*Figura 6.2: Diagrama de Despliegue*

### *6.1.2.1 Servidor 1*

Se trata de un servidor en el que se despliega un contenedor Docker, que tiene instalado Python 3 y Java JDK 8, además de contener la librería *mallet*, el dataset *20_newsgroups* y el fichero con los vectores de palabras de 100 dimensiones generados con *Glove*. Además de esto, contiene los siguientes elementos:

#### 6.1.2.1.1 Topics and Summary

Se trata del subsistema *Topics and Summary* identificado en la fase de análisis, que permite la comunicación con los modelos de extracción de topics y generación de resúmenes de textos. Está instalado dentro del contenedor Docker como una librería Python, y es utilizado por el subsistema *Web backend*.

#### 6.1.2.1.2 Web backend

Se trata del subsistema *Web backend* identificado en la fase de análisis, que se comunica con el subsistema *Topics and Summary* para obtener la funcionalidad de los modelos, y ofrece una API REST que expone dicha funcionalidad. Está instalado dentro del contenedor Docker como una librería Python, y utiliza *waitress* como servidor web.

Es utilizado por el subsistema *Web frontend*, pero en el futuro podría ser utilizado por otros subsistemas que consuman su API REST, como, por ejemplo, una aplicación móvil.

#### 6.1.2.2 Servidor 2

Se trata de un servidor en Heroku, en el que se despliega un contenedor Docker, que tiene instalado Node.js 10 y Angular 7, además del siguiente elemento:

##### 6.1.2.2.1 Web frontend

Se trata del subsistema *Web frontend* identificado en la fase de análisis, que se comunica con el subsistema *Web backend*, consumiendo su API REST, y ofrece una interfaz web que permite a los usuarios interaccionar con el sistema. Los archivos de la compilación de la aplicación Angular están presentes en el contenedor Docker, y se utiliza nginx como servidor web.

Es accedido por los usuarios, a través de un navegador web, en la siguiente url: http://topics-and-summary-web-front.herokuapp.com.

#### 6.1.2.3 Ventajas de realizar el despliegue en 2 servidores distintos

Tener en un servidor los subsistemas *Topics and Summary* y *Web backend* y en otro el subsistema *Web frontend* reporta varias ventajas:

- Permite escalar de forma independiente cada uno de los servidores.
- Incrementa la disponibilidad, dado que la caída del servidor del subsistema *Web frontend* no afecta a los otros 2 subsistemas, que pueden seguir respondiendo peticiones de futuros subsistemas que consuman la API REST.

## 6.2 Diseño de Clases

### 6.2.1 Diagrama de Clases

#### 6.2.1.1 Subsistema Topics and Summary

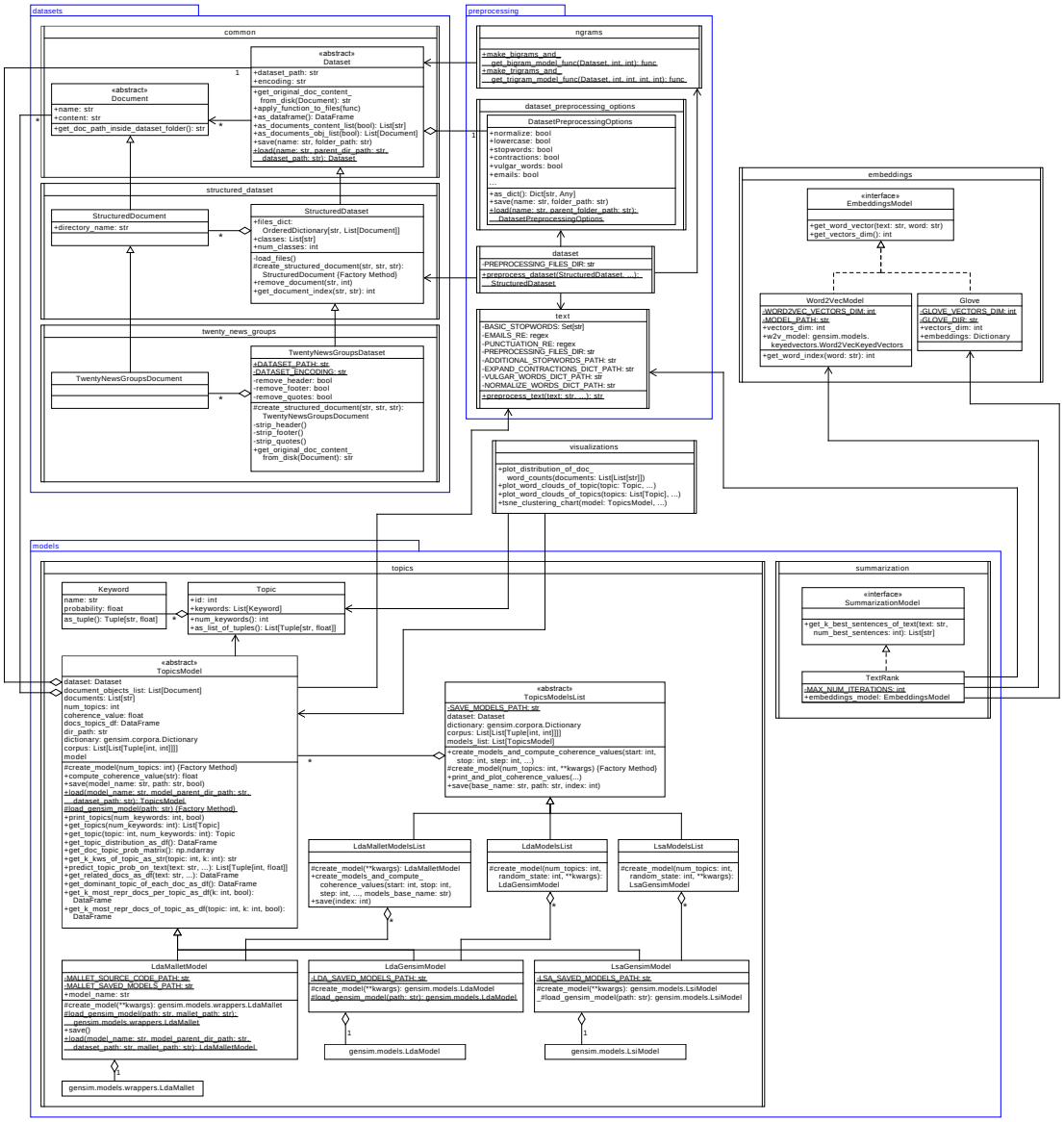



*Figura 6.3: Diagrama de Clases del subsistema Topics and Summary*

### 6.2.1.2 Subsistema Web backend

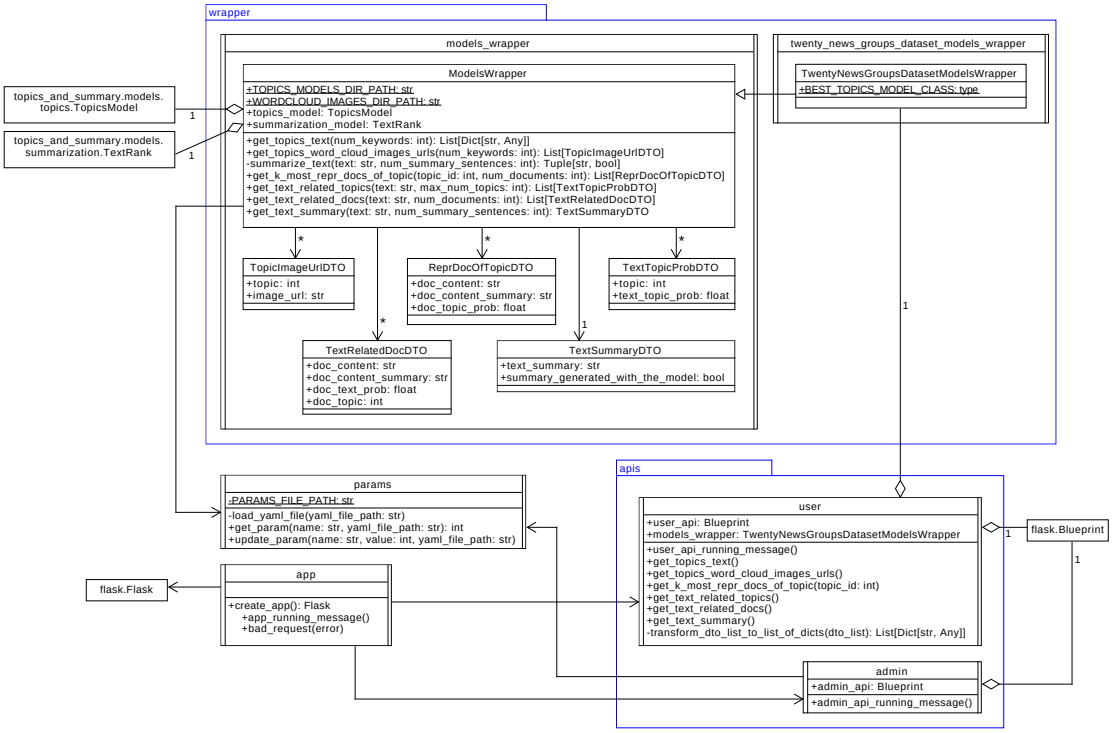



*Figura 6.4: Diagrama de Clases del subsistema Web backend*

### 6.2.1.3 Subsistema Web frontend

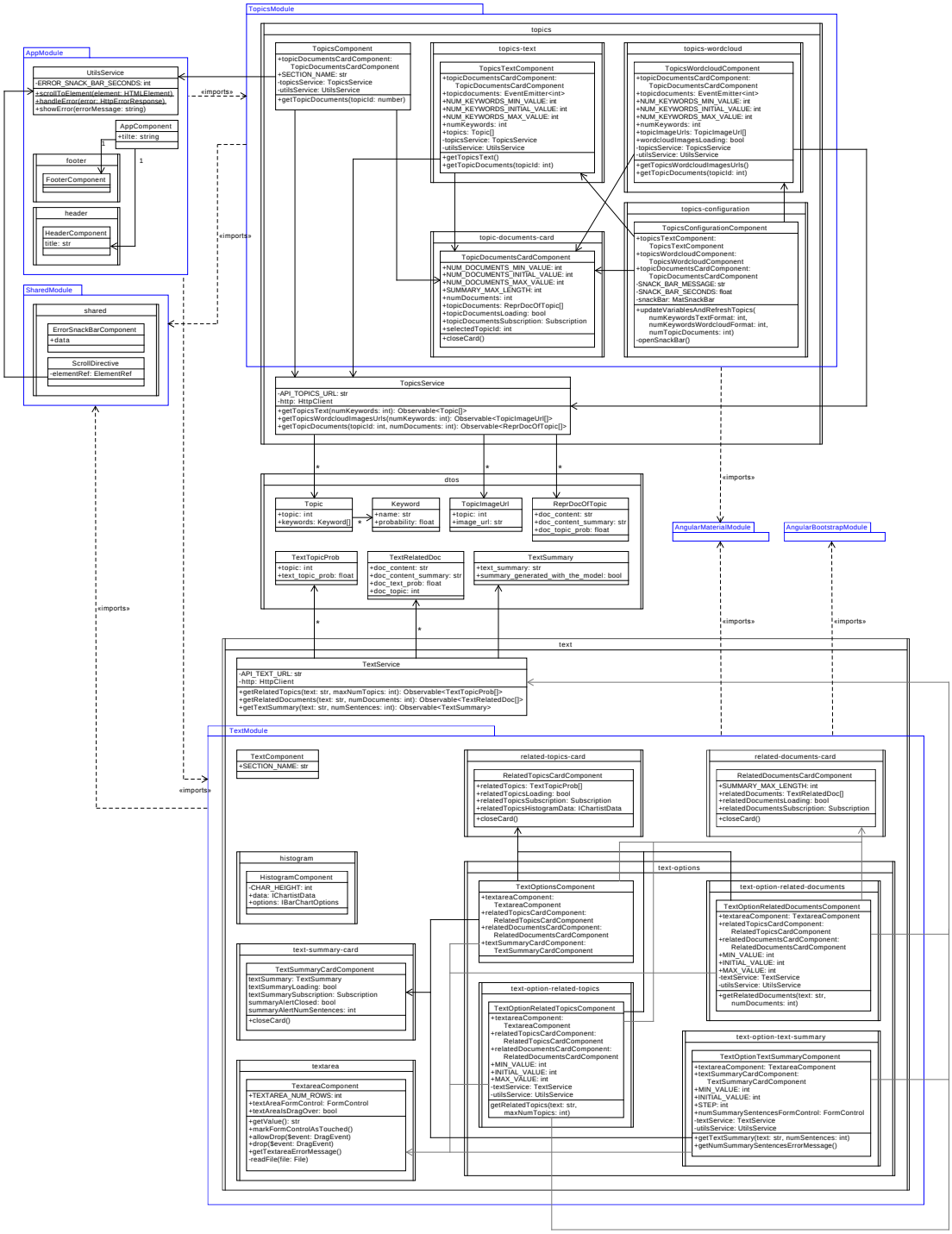



*Figura 6.5: Diagrama de Clases del subsistema Web frontend*

## 6.3 Diseño de la Interfaz

A continuación, se muestran los prototipos detallados de todas las pantallas:

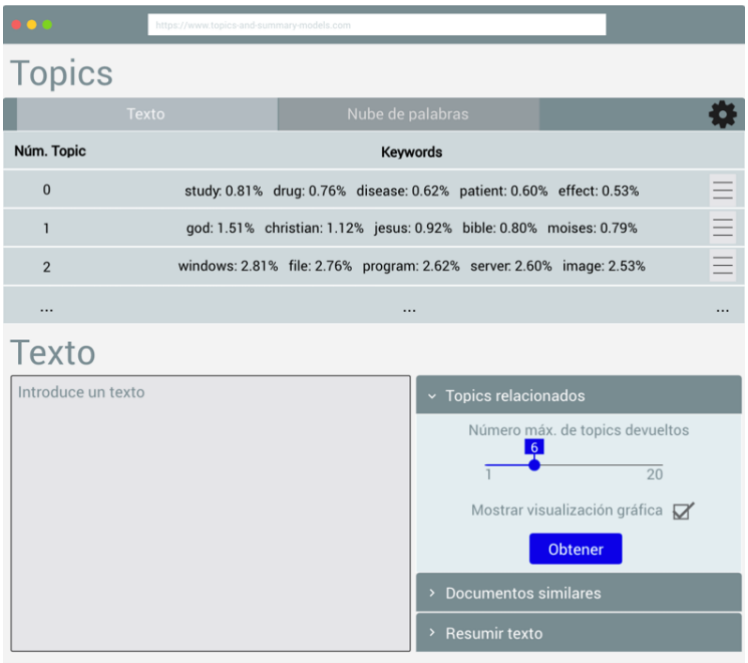



*Figura 6.6: Prototipo pantalla – Topics en formato texto*

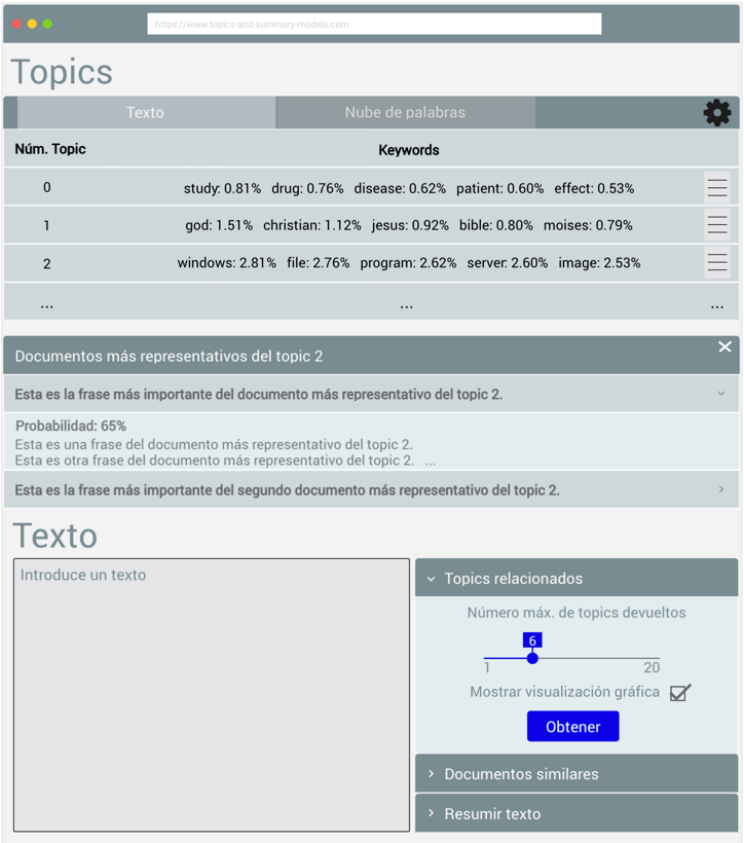



*Figura 6.7: Prototipo pantalla – Topics en formato texto y documentos más representativos*

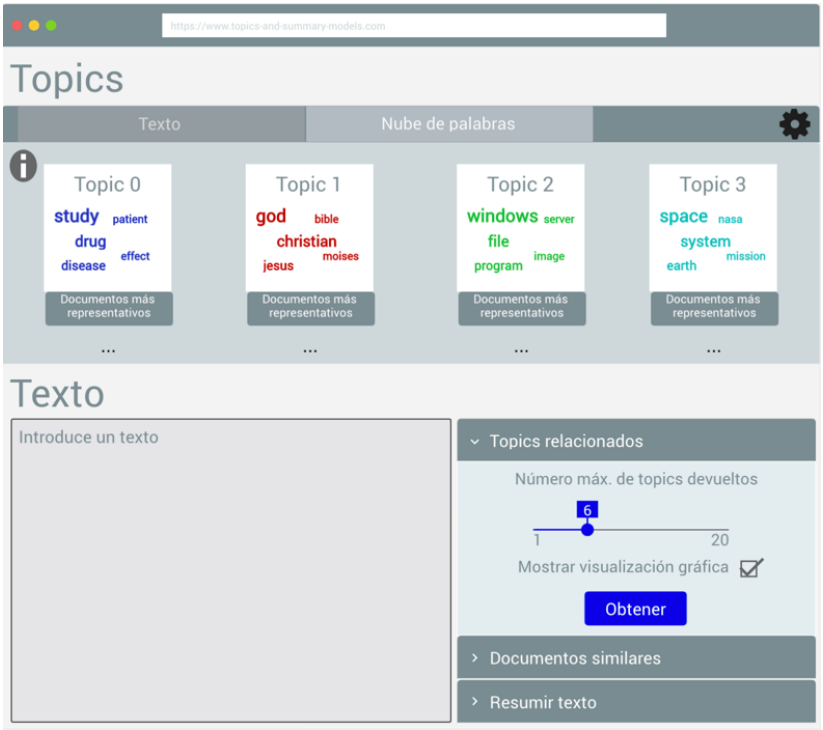



*Figura 6.8: Prototipo pantalla – Topics en formato nube de palabras*

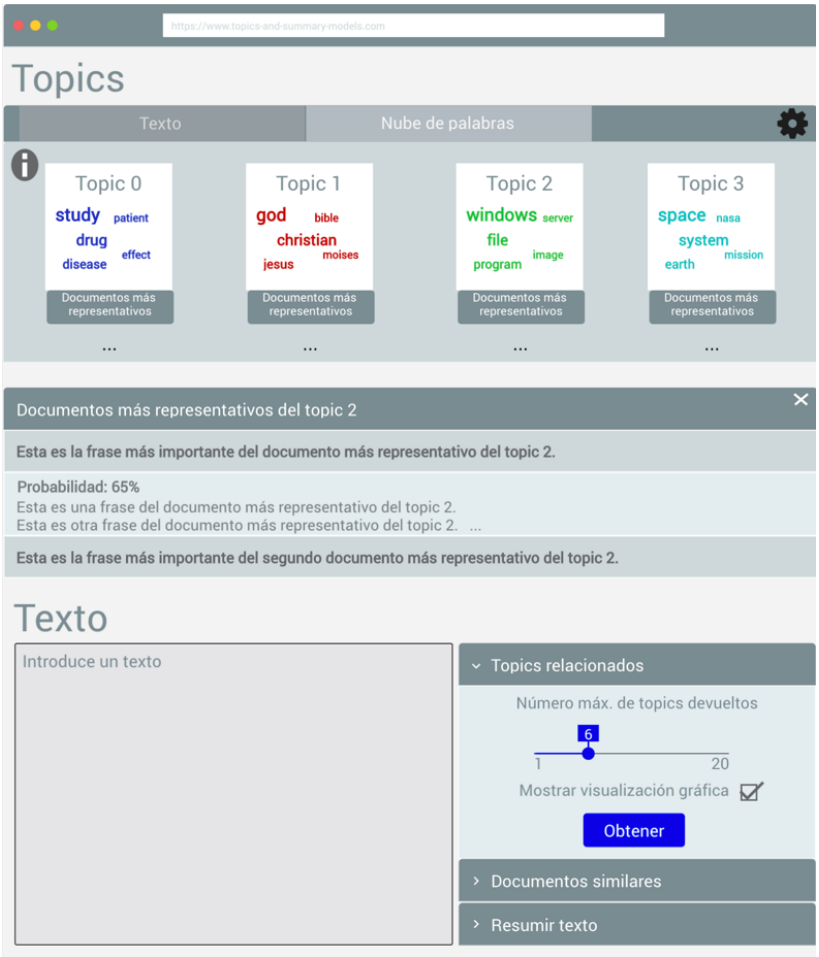



*Figura 6.9: Prototipo pantalla – Topics en formato nube de palabras y documentos más repr.*

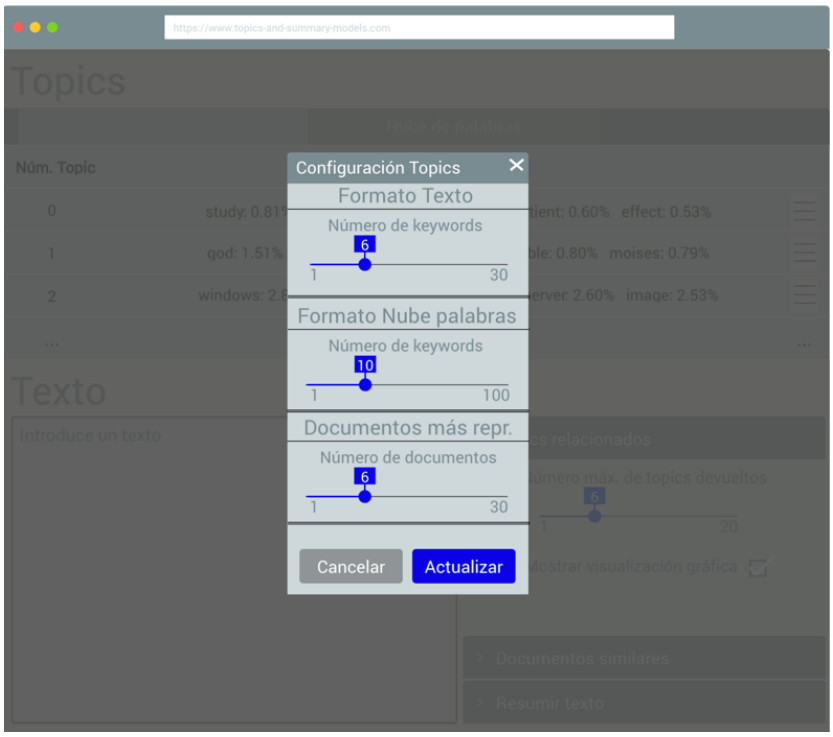



*Figura 6.10: Prototipo pantalla – Configuración de la visualización de topics*

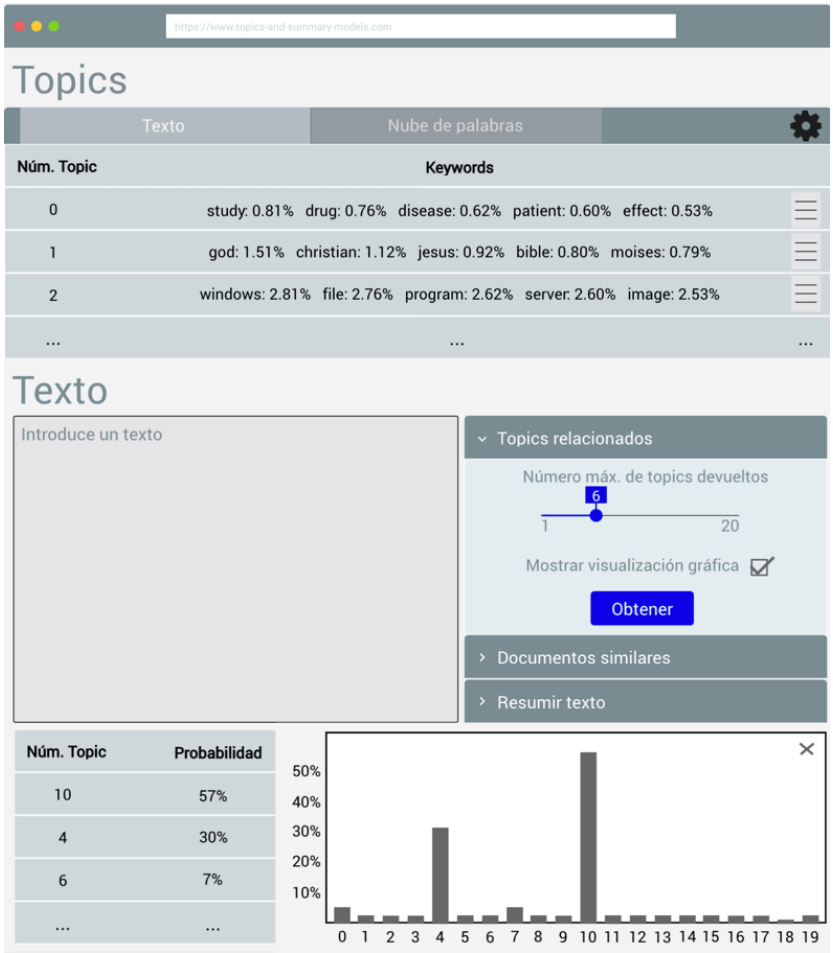



*Figura 6.11: Prototipo pantalla – Obtención de topics relacionados, dado un texto*

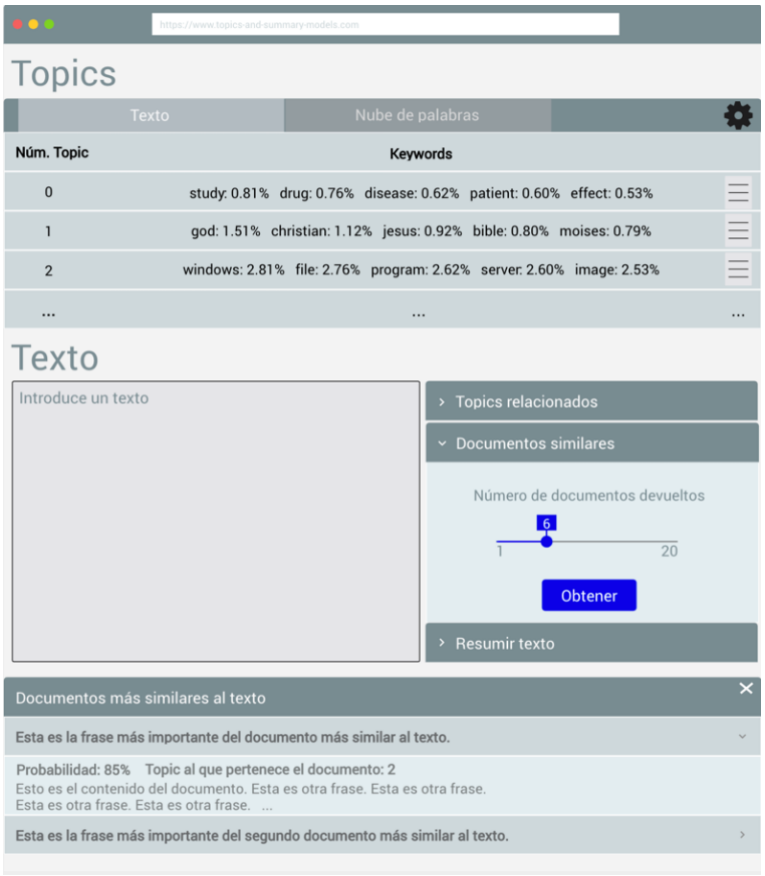



*Figura 6.12: Prototipo pantalla – Obtención de documentos similares, dado un texto*

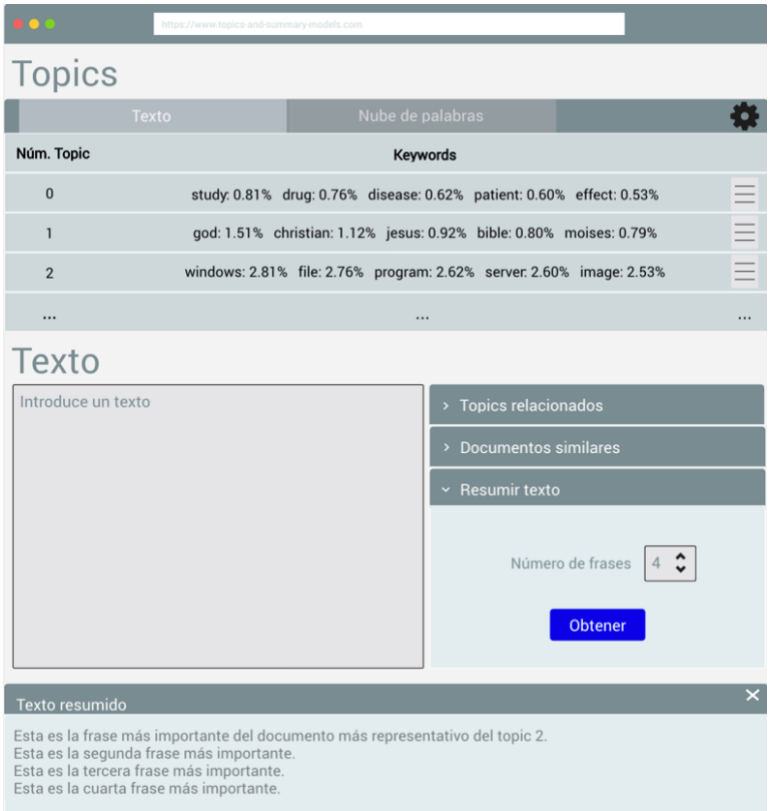



*Figura 6.13: Prototipo pantalla – Obtención de un resumen, dado un texto*

El siguiente enlace permite visualizar los prototipos anteriores de forma interactiva: https://www.fluidui.com/editor/live/preview/cF9pZnBBaVpXNjRPa0o2T1NaTE1wSGVJZjUyRlNhd09CNQ==. Los prototipos han sido realizados con la herramienta FluidUI [17].

## 6.4 Especificación Técnica del Plan de Pruebas

### 6.4.1 Pruebas Unitarias automatizadas

A continuación, se describen las funcionalidades que se probarán mediante las pruebas unitarias automatizadas que se realizarán en los subsistemas *Topics and Summary* y *Web backend*. Dentro de cada subsistema, se agruparán las funcionalidades por módulos Python (ficheros .py) o paquetes Python (carpetas que contienen un fichero __init__.py).

#### *6.4.1.1 Subsistema Topics and Summary*

##### 6.4.1.1.1 Paquete datasets

- Crear un objeto Dataset
- Obtener el texto original de uno de los documentos del objeto Dataset
- Transformar un objeto Dataset en una lista con los textos de los documentos
- Guardar un objeto Dataset en disco y cargarlo (con y sin las opciones de preprocesado)

##### 6.4.1.1.2 Paquete preprocessing

- Preprocesar un texto
- Preprocesar un objeto Dataset
- Crear un objeto DatasetPreprocessingOptions y transformarlo en un diccionario
- Guardar un objeto DatasetPreprocessingOptions en disco y cargarlo

##### 6.4.1.1.3 Módulo models.summarization

- Resumir un texto

##### 6.4.1.1.4 Módulo models.topics

- Guardar un objeto LdaGensimModel en disco y cargarlo
- Guardar un objeto LdaMalletModel en disco y cargarlo
- Guardar un objeto LsaGensimModel en disco y cargarlo
- Obtener los topics en formato texto de un objeto TopicsModel
- Obtener la probabilidad de un texto de estar relacionado con los topics de un objeto TopicsModel
- Obtener el topic más representativo de cada documento del dataset usando un objeto LdaGensimModel (aquí se utiliza en concreto un objeto LdaGensimModel, ya que es mucho más rápido que LdaMalletModel)
- Obtener los textos más similares a un texto dado, usando un objeto TopicsModel
- Obtener los k documentos más representativos de cada topic, usando un objeto TopicsModel

- Obtener los k documentos más representativos de un topic en concreto, usando un objeto TopicsModel

##### 6.4.1.1.5 Módulo utils

- Unir varios paths, con y sin separador al final de cada path, utilizando la barra normal como separador (/)
- Guardar un objeto cualquiera en disco y cargarlo
- Guardar una función cualquiera en disco y cargarla

#### *6.4.1.2 Subsistema Web backend*

##### 6.4.1.2.1 Módulo params

- Obtener parámetros del fichero yaml que contiene los parámetros del sistema
- Modificar parámetros del fichero yaml que contiene los parámetros del sistema

##### 6.4.1.2.2 Paquete wrapper

- Resumir un texto para el cuál el algoritmo TextRank no converge
- Resumir un texto para el cuál el algoritmo TextRank converge

### 6.4.2 Pruebas de Usabilidad

#### *6.4.2.1 Usuarios y metodología*

- **Usuarios:** Las pruebas de usabilidad se realizarán sobre 3 usuarios, con distintos perfiles y edades. Todos los usuarios desempeñarán el rol de "Usuario" descrito en la sección Identificación de Actores del Sistema.
- **Metodología:** Estas pruebas se basarán en dejar a los usuarios interaccionar con el sistema, así como en realizar un cuestionario de evaluación del usuario y un cuestionario para el responsable de las pruebas, que serán detallados en la siguiente sección. Se rellenarán ambos cuestionarios para cada uno de los 3 usuarios. Una vez terminados, se analizarán conjuntamente los resultados de las pruebas para establecer una conclusión respecto a los siguientes tres aspectos de usabilidad:
    - **Facilidad de aprendizaje:** Capacidad para aprender la funcionalidad del sistema y desarrollar las tareas de manera adecuada, midiendo cuánto se tarda en hacer las distintas tareas.
    - **Errores:** Cuántos errores cometen los usuarios en las distintas tareas, lo que decrementa la usabilidad del sistema.
    - **Satisfacción del usuario:** Impresión general de los usuarios al usar el sistema.

  Por último, se establecerán unas conclusiones, que dictarán el camino a seguir para mejorar el sistema en cuanto a la usabilidad, y se indicará qué cambios se han realizado en la interfaz en base a dichas conclusiones.

#### 6.4.2.2 Cuestionarios

##### 6.4.2.2.1 Cuestionario de Evaluación

Este cuestionario será rellenado utilizando la información proporcionada por los usuarios.

Se divide en 4 partes:

1. **Preguntas de carácter general** a través de las cuales se determinará el tipo de usuario, su nivel de conocimiento informático y su nivel de inglés.
2. **Actividades guiadas** que realizará el usuario con el sistema.
3. **Batería de preguntas cortas** sobre determinados aspectos del sistema.
4. **Observaciones**, en las cuáles el usuario aportará todo lo que considere oportuno del sistema.

*6.4.2.2.1.1 Preguntas de carácter general*

| **¿Utiliza un ordenador, un dispositivo móvil o una tablet frecuentemente?** |
|---|
| 1. Todos los días<br>2. Varias veces a la semana<br>3. Ocasionalmente<br>4. Nunca o casi nunca |
| **¿Qué tipo de actividades realiza con el ordenador / dispositivo móvil / tablet?** |
| 1. Es parte de mi trabajo o profesión<br>2. Lo uso básicamente para ocio<br>3. Solo empleo aplicaciones estilo Office<br>4. Únicamente leo el correo y navego ocasionalmente |
| **¿Ha usado alguna vez software como el de esta prueba?** |
| 1. Sí, he empleado software similar<br>2. No, aunque sí empleo otros programas que me ayudan a realizar tareas similares<br>3. No, nunca |
| **¿Qué busca Vd. Principalmente en un programa?** |
| 1. Que sea fácil de usar<br>2. Que sea intuitivo<br>3. Que sea rápido<br>4. Que tenga todas las funciones necesarias |
| **¿Qué edad tiene?** |
| |
| **¿Cuál es su profesión?** |
| |

| ¿Cómo definiría su nivel de inglés? |
|---|
| 1. Bajo<br>2. Medio<br>3. Alto |

##### 6.4.2.2.1.2 Actividades guiadas

La lista de actividades a realizar por el usuario es la siguiente:

- Consultar los topics del dataset en formato texto
- Consultar los topics del dataset en formato nube de palabras
- Consultar las keywords de un determinado topic, identificar cuáles tienen más importancia dentro del mismo, y tratar de averiguar la temática del topic
- Obtener los documentos más representativos de un topic
- Configurar el número de keywords mostradas para los topics del dataset en ambos formatos, así como el número de documentos más representativos de un topic devueltos, y verificar que se ha actualizado el sistema en base a dicha configuración
- Obtener los topics más relacionados con un texto introducido por el usuario
- Obtener el documento (uno solamente) más similar a un texto introducido por el usuario
- Obtener un resumen de 3 frases de un texto introducido por el usuario

El usuario deberá comentar los problemas y dificultades que, según su opinión, presenta el sistema. Estas opiniones serán recogidas en el presente documento.

##### 6.4.2.2.1.3 Preguntas Cortas sobre la Aplicación y Observaciones

| **Facilidad de Uso** | **Siempre** | **Frecuentemente** | **Ocasionalmente** | **Nunca** |
|---|---|---|---|---|
| *¿Sabe donde está dentro de la aplicación?* | | | | |
| *¿Existe ayuda para las funciones en caso de que tenga dudas?* | | | | |
| *¿Le resulta sencillo el uso de la aplicación?* | | | | |
| **Funcionalidad** | **Siempre** | **Frecuentemente** | **Ocasionalmente** | **Nunca** |
| *¿Funciona cada tarea como Vd. Espera?* | | | | |
| *¿Es capaz de identificar la temática de los topics?* | | | | |
| *¿Los documentos más representativos de cada topic son coherentes con la temática del mismo?* | | | | |
| *¿El topic con mayor probabilidad de estar relacionado con el texto que ha introducido es coherente, respecto a la temática del texto?* | | | | |
| *¿El resumen del texto que ha introducido es bueno (es representativo de dicho texto)?* | | | | |

| | | | | |
|---|---|---|---|---|
| *¿El tiempo de respuesta de la aplicación es muy grande?* | | | | |
| **Calidad del Interfaz** | | | | |
| **Aspectos gráficos** | **Muy Adecuado** | **Adecuado** | **Poco Adecuado** | **Nada Adecuado** |
| *El tipo y tamaño de letra es* | | | | |
| *Los iconos e imágenes usados son* | | | | |
| *Los colores empleados son* | | | | |
| **Diseño de la Interfaz** | | **Sí** | **No** | **A veces** |
| *¿Le resulta fácil de usar?* | | | | |
| *¿El diseño de las pantallas es claro y atractivo?* | | | | |
| *¿Cree que el programa está bien estructurado?* | | | | |
| **Observaciones** | | | | |
| Cualquier comentario del usuario | | | | |

#### 6.4.2.2.2 Cuestionario para el Responsable de las Pruebas

En el siguiente cuestionario, el responsable de las pruebas anotará los distintos aspectos que observe durante la realización de las pruebas.

| **Aspecto Observado** | **Notas** |
|---|---|
| *Tiempo en realizar cada tarea* | Normal o Alto |
| *Errores cometidos* | 0, 1, 2, … |
| *<Otros aspectos>* | … |

### 6.4.3 Pruebas de Adaptabilidad

Parar probar que la interfaz web del sistema es adaptable se utilizará el navegador web Google Chrome, que permite visualizar la página web con distintos tamaños de pantalla.

Para superar estas pruebas, debe ser posible utilizar todas las funcionalidades de la página web correctamente en todos los tamaños de pantalla.

# Capítulo 7. Implementación del Sistema

## 7.1 Estándares y Normas Seguidos

Para los subsistemas *Topics and Summary* y *Web backend*, cuya implementación se ha realizado en Python, se ha seguido la ***Style Guide for Python Code*** [18], mayormente conocida como **PEP8** (nombre que viene de ser la *Python Enhancement Proposal* número 8). El cumplimiento de este guía se ha realizado siguiendo las advertencias del IDE, que está configurado para inspeccionar que el código cumple con PEP8.

Para el subsistema *Web frontend*, cuya implementación se ha realizado en Angular, se ha intentado seguir lo más fielmente posible la **guía de estilo** propuesta en la documentación de **Angular** [19].

## 7.2 Lenguajes de Programación

### 7.2.1 Python 3

- **Versión:** 3.6.7
- **Distribución:** Anaconda
- **Gestores de entornos:**
    - **Conda 4.6.14**
    - **Virtualenv 16.0.0**
- **Librerías utilizadas:**
    - **Gensim 3.4.0**
    - **Networkx 2.2**
    - Numpy 1.15.4
    - **Pandas 0.24.1**
    - **NLTK 3.3**
    - Scikit-learn 0.20.3
    - **Matplotlib 3.0.2**
    - Seaborn 0.9.0
    - **Wordcloud 1.5.0**
    - Bokeh 1.0.4
    - **Dill 0.2.9**
    - **Typing 3.6**
    - **Setuptools 40.6.3**
    - Texttable 1.6.1
    - Tqdm 4.31.1
    - **Sphinx 2.0.1**
    - Pipreqs 0.4.9
    - **Flask 1.0.2**
    - Werkzeug 0.15.2
    - **Flask-Cors 3.0.7**
    - **PyYAML 5.1**

### 7.2.2 TypeScript

- **Versión:** 3.2.2
- **Principales librerías y frameworks utilizados:**
    - **Angular**
        - **Core 7.2.0**
        - **Material 7.3.7**
        - Common 7.2.0
        - Forms 7.2.0
    - **Bootstrap 4.3.1**
    - **NG Bootstrap 4.2.1**

- Chartist 0.11.2
- NG Chartist 4.1.0
- **Rxjs 6.3.3**

### 7.2.3 HTML

- **Versión:** 5

### 7.2.4 CSS

- **Versión:** 4

## 7.3 Herramientas y Programas Usados para el Desarrollo

### 7.3.1 PyCharm

**Versión:** 2019.1.3 (Professional Edition)

**Página web:** https://www.jetbrains.com/pycharm

Se trata de un IDE para desarrollar en el lenguaje Python.

Ha sido utilizado para implementar los subsistemas *Topics and Summary* y *Web backend*.

### 7.3.2 WebStorm

**Versión:** 2019.1.3

**Página web:** https://www.jetbrains.com/webstorm

Se trata de un IDE para desarrollo web.

Ha sido utilizado para implementar el subsistema *Web frontend*.

### 7.3.3 Postman

**Versión:** 7.2.2

**Página web:** https://www.getpostman.com

Se trata de una aplicación para facilitar el desarrollo de APIs.

Ha sido utilizado para probar la API REST del subsistema *Web backend*.

### 7.3.4 Git

**Versión:** 2.20.1

**Página web:** https://git-scm.com

Se trata de un sistema de control de versiones distribuido.

Ha sido utilizado para mantener un control de versiones de los subsistemas.

### 7.3.5 Docker

**Versión:** Docker Desktop Community 2.0.0.3

**Página web:** https://www.docker.com

Se trata de una plataforma para desarrolladores y administradores de sistemas, basada en contenedores.

Ha sido utilizado para empaquetar los 3 subsistemas: En el caso del subsistema *Topics and Summary*, para ejecutar una demo que muestra su funcionalidad en un entorno aislado; y en el caso de los subsistemas *Web backend* y *Web frontend*, para permitir su ejecución en entornos aislados, y facilitar su despliegue.

### 7.3.6 Vagrant

**Versión:** 2.2.5

**Página web:** https://www.vagrantup.com

Se trata de una herramienta para la creación y manejo de máquinas virtuales.

Ha sido utilizado para crear una máquina virtual, dentro de la cual se genera una imagen Docker del subsistema *Topics and Summary*, para poder así ejecutar la demo de dicho subsistema sin necesidad de tener instalado Docker en la máquina host.

### 7.3.7 Chrome

**Versión:** 75.0.3770.100

**Página web:** https://www.google.com/intl/es/chrome/

Se trata de un navegador web desarrollado por Google.

Ha sido utilizado para probar y depurar el subsistema *Web frontend,* incluyendo las pruebas de adaptabilidad de la interfaz web.

# 7.4 Creación del Sistema

## 7.4.1 Problemas Encontrados

### 7.4.1.1 Rutas en Unix y Windows

#### 7.4.1.1.1 Explicación del problema

El desarrollo del subsistema Topics and Summary se realizó en un equipo con sistema operativo MacOS, por lo que, inicialmente, las rutas que hacían referencia a carpetas o ficheros se incluyeron del siguiente modo:

```
path = "path/to/file"
```

Es decir, utilizando como separador barras tipo Unix. En este equipo, el código funcionaba correctamente, pero al probarlo en un equipo con sistema operativo Windows, las rutas comenzaron a fallar, dado que en Windows se utilizan barras invertidas "\" como separador.

#### 7.4.1.1.2 Solución

La solución fue utilizar una función de la librería estándar de Python, a la que se le pasa una ruta tipo Unix, y devuelve la ruta en el formato adecuado para el sistema operativo sobre el que se está ejecutando el programa Python. Es decir, si es un sistema operativo Unix, los separadores se mantienen como barras normales "/", mientras que, si es Windows, las barras "/" se cambian por barras invertidas "\".

### 7.4.1.2 Problema con los imports a la hora de generar un package en Python

#### 7.4.1.2.1 Explicación del problema

Al principio del desarrollo, los imports que hacían referencia a módulos del propio subsistema *Topics and Summary* estaban puestos como imports absolutos, pero sin indicar el nombre del proyecto, es decir, de la siguiente forma:

```
from datasets.common import get_file_content, Document, Dataset
```

Durante el desarrollo, esto no suponía ningún problema, dado que al realizar pruebas desde la consola de Python en el propio IDE se tenía acceso directamente a todos los módulos y paquetes Python. Sin embargo, al instalar el subsistema *Topics and Summary* como una librería para que pudiese ser utilizada por el subsistema *Web backend*, estos imports comenzaron a fallar, como se muestra en la siguiente imagen:

```
In [5]: from topics_and_summary.datasets.twenty_news_groups import TwentyNewsGroupsDataset
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
<ipython-input-5-77cfed35d437> in <module>
----> 1 from topics_and_summary.datasets.twenty_news_groups import TwentyNewsGroupsDataset

~/Desktop/Clase/4/2_Cuatri/TFG/Codigo/topics_and_summary/topics_and_summary/datasets/twenty_news_groups.py in <module>
      7 import pandas as pd
      8
----> 9 from datasets.common import get_file_content, Document, Dataset
     10 from utils import pretty_print, get_abspath_from_project_root, join_paths
     11

ModuleNotFoundError: No module named 'datasets.common'
```

El motivo del fallo se debe a que Python no es capaz de encontrar el paquete *datasets*, dado que para resolver los imports, Python comprueba en los siguientes sitios:

1. En los módulos *built-in* de Python, es decir, de la librería estándar.
2. En el directorio desde el que se ejecuta el script actual.
3. En los directorios especificados en la variable de entorno PYTHONPATH.
4. En otros directorios indicados en la instalación de Python.

El motivo por el que desde el IDE esta forma de hacer los imports funciona correctamente es porque el propio IDE ejecuta la siguiente línea en la consola de Python, al principio de la ejecución:

```
sys.path.extend(['/ruta/hasta/carpeta/proyecto/topics_and_summary'])
```

Donde */ruta/hasta/carpeta/proyecto/* es la ruta donde está la raíz del proyecto. Esto lo que hace es añadir dicha ruta al PYTHONPATH y, por tanto, consigue que se pueda acceder a los módulos del proyecto directamente, sin necesidad de incluir delante el nombre del proyecto.

Cuando se instala el proyecto como una librería y se intenta utilizar, al no haber nada como lo anterior que indique donde están los módulos del proyecto, Python no es capaz de encontrarlos.

#### 7.4.1.2.2 Solución

La solución a esto es cambiar los imports en el código fuente (tanto en los módulos que tienen la funcionalidad como en los módulos que tienen los tests), indicando en ellos el nombre del proyecto, de la siguiente forma:

```
from topics_and_summary.datasets.common import get_file_content, Document, Dataset
```

De este modo, podemos seguir accediendo a los módulos desde el IDE, ya que se sigue extendiendo el PYTHONPATH, y también podemos acceder a los módulos desde la librería instalada en nuestro equipo, dado que al indicar al comienzo del import el nombre de la librería, Python es capaz de saber donde está dicha librería instalada, y puede acceder a sus módulos.

#### 7.4.1.3 Versión de la librería gensim

##### 7.4.1.3.1 Explicación del problema

Después de instalar el subsistema *Topics and Summary* como una librería Python en un virtualenv aislado dentro de mi entorno de desarrollo, la demo presente en dicho subsistema comenzó a fallar. Sin embargo, al ejecutar la demo desde el IDE, ésta funcionaba correctamente.

Tras revisar la traza del error e indagar en el código fuente de la librería gensim, me di cuenta de que el fallo se daba al cargar de disco una lambda expresión previamente serializada, cuyo cuerpo de la función realizaba llamadas a objetos de la clase gensim.models.phrases.Phraser de gensim.

El fallo se debía a que en el fichero setup.py, dentro del parámetro install_requires de la función setup, había establecido que las versiones de las dependencias que utilizaba el subsistema tenían que ser iguales o superiores a las indicadas, tal que así:

```
install_requires=[
          'numpy>=1.15.4',
          'gensim>=3.4.0',
          ...
]
```

Por tanto, a la hora de instalarse el subsistema *Topics and Summary*, se instalaba siempre la última versión disponible de sus dependencias, dado que la única restricción es que dicha versión fuese mayor o igual que la establecida en install_requires.

Esto causó que en el virtualenv donde había instalado *Topics and Summary* se instalase la versión 3.7.3 de gensim, en lugar de la 3.4.0, que era con la que estaba desarrollando. Al inspeccionar el código fuente de ambas versiones de gensim, me di cuenta de que en la versión 3.7.3 el método que me estaba fallando tenía una implementación ligeramente distinta (en concreto, antes se obtenía el segundo elemento de una variable usando el operador *[1]*, mientras que en la versión 3.7.3 se devolvía la variable).

Una explicación más en detalle de la traza de error, el método que la provocaba, y las diferencias de dicho método en ambas versiones se puede encontrar en el siguiente Issue en GitHub: https://github.com/RaRe-Technologies/gensim/issues/2517. Este Issue lo abrí para comentar a los desarrolladores de gensim el fallo que me había encontrado, pensando que podía tratarse de un bug. El creador de la librería me aclaró que estaba controlado, y que si utilizaba los métodos para guardar y cargar el objeto gensim.models.phrases.Phraser (los cuales habían sido modificados para asegurar la compatibilidad hacia atrás) no tendría problemas con el cambio que realizaron en dicho método.

##### 7.4.1.3.2 Solución

La solución que proponía el autor de gensim era la más correcta, pero también la más rompedora, ya que implicaba modificar gran parte de cómo se guardaba y cargaba la función que genera ngrams, dado que en lugar de simplemente serializar esa función, tendría que

guardar los objetos gensim.models.phrases.Phraser y luego reconstruir la función. Además, no podría tener un método genérico que permita guardar y cargar modelos que generen bigrams o trigrams, dado que el modelo que genera trigrams utiliza dos objetos gensim.models.phrases.Phraser en lugar de uno, tal y como se puede ver en el siguiente fragmento de código:

```
# Modelo de bigrams
bigram_model = gensim.models.phrases.Phraser(bigram)
# Modelo de trigrams
trigram_model = gensim.models.phrases.Phraser(trigram)

# Función lambda que permite generar bigrams
bigram_model_func = lambda x: bigram_model[x]

# Función lambda que permite generar trigrams
trigram_model_func = lambda x: trigram_model[bigram_model[x]]
```

Por tanto, la solución final adoptada fue mantener la forma en la que se estaba guardando dicha función, y modificar el fichero setup.py para especificar las versiones de las dependencias de forma restrictiva, es decir, instalar siempre la misma versión que la utilizada en desarrollo, para evitar así posibles problemas como el que se ha dado en este caso.

El parámetro install_requires de la función setup finalmente quedó así:

```
install_requires=[
          'numpy==1.15.4',
          'gensim==3.4.0',
          ...
]
```

## 7.4.2 Descripción Detallada de las Clases

A continuación, se indican las rutas donde se encuentra la documentación del código fuente de cada uno de los 3 subsistemas, dentro del Archivo adjunto.

### *7.4.2.1 Subsistema Topics and Summary*

**Ruta:** topics_and_summary/topics_and_summary/docs/build/html/api-reference.html

**Herramienta utilizada para generar la documentación:** Sphinx

### *7.4.2.2 Web backend*

**Ruta:** web_backend/web_backend/docs/build/html/api-reference.html

**Herramienta utilizada para generar la documentación:** Sphinx

### *7.4.2.3 Web frontend*

**Ruta:** web-frontend/documentation/modules.html

**Herramienta utilizada para generar la documentación:** Compodoc

# Capítulo 8. Desarrollo de las Pruebas

## 8.1 Pruebas Unitarias automatizadas

A continuación, se describen en detalle las pruebas unitarias automatizadas que se han implementado parar probar las funcionalidades descritas en el apartado 6.4.1. En todas ellas, el resultado obtenido coincide con el resultado esperado.

En muchas pruebas, lo que se comprueba es que el resultado actual de un determinado método coincide con el resultado obtenido previamente utilizando ese mismo método (serializando para ello dicho resultado en disco). Esto se debe a que, dada la complejidad y subjetividad de gran parte de la aplicación (siendo en algunos casos incluso no determinista, como en la generación de algunos modelos), la manera más sencilla de probar muchas de las funcionalidades es ejecutándolas, comprobando a mano que sus resultados son correctos, y almacenando dichos resultados, para posteriormente basar las pruebas en ellos. Así, se puede verificar que cambios futuros no estropean funcionalidades ya implementadas y que funcionaban correctamente.

### 8.1.1 Subsistema Topics and Summary

| ***Paquete datasets*** | |
|---|---|
| **Prueba** | **Resultado Esperado** |
| Crear un objeto Dataset | El objeto se crea sin errores, y es igual que un objeto creado previamente y guardado en disco |
| **Prueba** | **Resultado Esperado** |
| Obtener el texto original de uno de los documentos del objeto Dataset | El texto devuelto coincide con el texto original de dicho documento |
| **Prueba** | **Resultado Esperado** |
| Transformar un objeto Dataset en una lista con los textos de los documentos | La lista devuelta es igual que una lista con los textos de los documentos previamente calculada y guardada en disco |
| **Prueba** | **Resultado Esperado** |
| Guardar un objeto Dataset en disco y cargarlo (sin las opciones de preprocesado) | Se guarda en disco el objeto Dataset (sin tener un objeto DatasetPreprocessingOptions asociado) usando la función save, y se carga ese mismo objeto de disco usando la función load, y ambos objetos deben ser iguales |
| **Prueba** | **Resultado Esperado** |
| Guardar un objeto Dataset en disco y cargarlo (con las opciones de preprocesado) | Se guarda en disco el objeto Dataset (con un objeto DatasetPreprocessingOptions asociado) usando la función save, y se carga ese mismo objeto de disco usando la función load, y ambos objetos deben ser iguales |

| ***Paquete preprocessing*** | |
|---|---|
| **Prueba** | **Resultado Esperado** |
| Preprocesar un texto | Se parte de un texto breve, sobre el que se aplican todas las opciones de preprocesado, y se espera que coincida con el texto esperado |
| **Prueba** | **Resultado Esperado** |
| Preprocesar un objeto Dataset | Se parte de un objeto Dataset, sobre el que se aplican todas las opciones de preprocesado, y se espera que coincida con un objeto Dataset previamente preprocesado con las mismas opciones y guardado en disco |
| **Prueba** | **Resultado Esperado** |
| Crear un objeto DatasetPreprocessingOptions y transformarlo en un diccionario | El diccionario contiene todas las opciones, y con los mismos valores que el objeto. Además, la función para generar ngrams almacenada en el diccionario genera los mismos ngrams que esa función almacenada en el objeto. |
| **Prueba** | **Resultado Esperado** |
| Guardar un objeto DatasetPreprocessingOptions en disco y cargarlo | Se guarda en disco el objeto DatasetPreprocessingOptions usando la función save, y se carga ese mismo objeto de disco usando la función load, y ambos objetos deben ser iguales, y sus funciones para generar ngrams deben generar los mismos ngrams |

| ***Módulo models.summarization*** | |
|---|---|
| **Prueba** | **Resultado Esperado** |
| Resumir un texto | El resumen devuelto es igual que un resumen generado previamente y guardado en disco |

| ***Módulo models.topics*** | |
|---|---|
| **Prueba** | **Resultado Esperado** |
| Guardar un objeto LdaGensimModel en disco y cargarlo | Se guarda en disco el objeto LdaGensimModel usando la función save, y se carga ese mismo objeto de disco usando la función load, y ambos objetos deben ser iguales |
| **Prueba** | **Resultado Esperado** |
| Guardar un objeto LdaMalletModel en disco y cargarlo | Se guarda en disco el objeto LdaMalletModel usando la función save, y se carga ese mismo objeto de disco usando la función load, y ambos objetos deben ser iguales |
| **Prueba** | **Resultado Esperado** |
| Guardar un objeto LsaGensimModel en disco y cargarlo | Se guarda en disco el objeto LsaGensimModel usando la función save, y se carga ese mismo objeto de disco usando la función load, y ambos objetos deben ser iguales |
| **Prueba** | **Resultado Esperado** |
| Obtener los topics en formato texto de un objeto TopicsModel | Los topics devueltos son iguales que los topics generados previamente y guardados en disco, usando ese mismo modelo |
| **Prueba** | **Resultado Esperado** |
| Obtener la probabilidad de un texto de estar relacionado con los topics de un objeto TopicsModel | Las probabilidades devueltas son iguales que las probabilidades obtenidas previamente y guardadas en disco, usando ese mismo modelo y ese mismo texto |

| Prueba | Resultado Esperado |
|---|---|
| Obtener el topic más representativo de cada documento del dataset usando un objeto LdaGensimModel | El dataframe que contiene el topic más representativo de cada documento es igual que el generado previamente y guardado en disco, usando ese mismo modelo |
| **Prueba** | **Resultado Esperado** |
| Obtener los textos más similares a un texto dado, usando un objeto TopicsModel | El dataframe que contiene los textos más similares a un texto dado es igual que el generado previamente y guardado en disco, usando ese mismo modelo y ese mismo texto |
| **Prueba** | **Resultado Esperado** |
| Obtener los k documentos más representativos de cada topic, usando un objeto TopicsModel | El dataframe que contiene los k documentos más representativos de cada topic es igual que el generado previamente y guardado en disco, usando ese mismo modelo |
| **Prueba** | **Resultado Esperado** |
| Obtener los k documentos más representativos de un topic en concreto, usando un objeto TopicsModel | El dataframe que contiene los k documentos más representativos de un topic en concreto es igual que el generado previamente y guardado en disco, usando ese mismo modelo |

| ***Módulo utils*** | |
|---|---|
| **Prueba** | **Resultado Esperado** |
| Unir varios paths, con y sin separador al final de cada path, utilizando la barra normal como separador (/) | Los paths se unen correctamente, utilizando el separador del sistema operativo donde se ejecuta |
| **Prueba** | **Resultado Esperado** |
| Guardar un objeto cualquiera en disco y cargarlo | Se guarda en disco el objeto usando la función save, y se carga ese mismo objeto de disco usando la función load, y ambos objetos deben ser iguales |
| **Prueba** | **Resultado Esperado** |
| Guardar una función cualquiera en disco y cargarla | Se guarda en disco la función usando la función save, y se carga esa misma función de disco usando la función load. Los resultados de invocar a ambas funciones deben coincidir. |

### 8.1.2 Subsistema Web backend

| ***Módulo params*** | |
|---|---|
| **Prueba** | **Resultado Esperado** |
| Obtener parámetros del fichero yaml que contiene los parámetros del sistema | Se obtiene el valor de varios parámetros de dicho fichero, y los valores obtenidos deben coincidir con los valores que tienen dichos parámetros en el fichero |
| **Prueba** | **Resultado Esperado** |
| Modificar parámetros del fichero yaml que contiene los parámetros del sistema | Se modifica el valor de varios parámetros de dicho fichero. Después, se obtiene el valor de esos mismos parámetros, y los valores obtenidos deben coincidir con los valores que se habían dado. Además, el resto de parámetros no modificados debe permanecer con sus mismos valores. |

| ***Paquete wrapper*** | |
|---|---|
| **Prueba** | **Resultado Esperado** |
| Resumir un texto para el cuál el algoritmo TextRank no converge | Se parte de un texto para el cuál se sabe que el algoritmo no converge, y se debe generar un texto de k frases, consistente en las k primeras frases del texto original, además de indicar que el resumen no ha sido generado con el modelo |
| **Prueba** | **Resultado Esperado** |
| Resumir un texto para el cuál el algoritmo TextRank converge | Se parte de un texto para el cuál se sabe que el algoritmo converge, y se debe generar un texto de k frases, que no consiste en las k primeras frases del texto original, además de indicar que el resumen ha sido generado con el modelo. |

## 8.2 Análisis estático de código

A continuación, se muestran los resultados del análisis estático de código, realizados con la herramienta SonarQube, para los subsistemas implementados en Python (*Topics and Summary* y *Web backend*).

### 8.2.1 Subsistema Topics and Summary

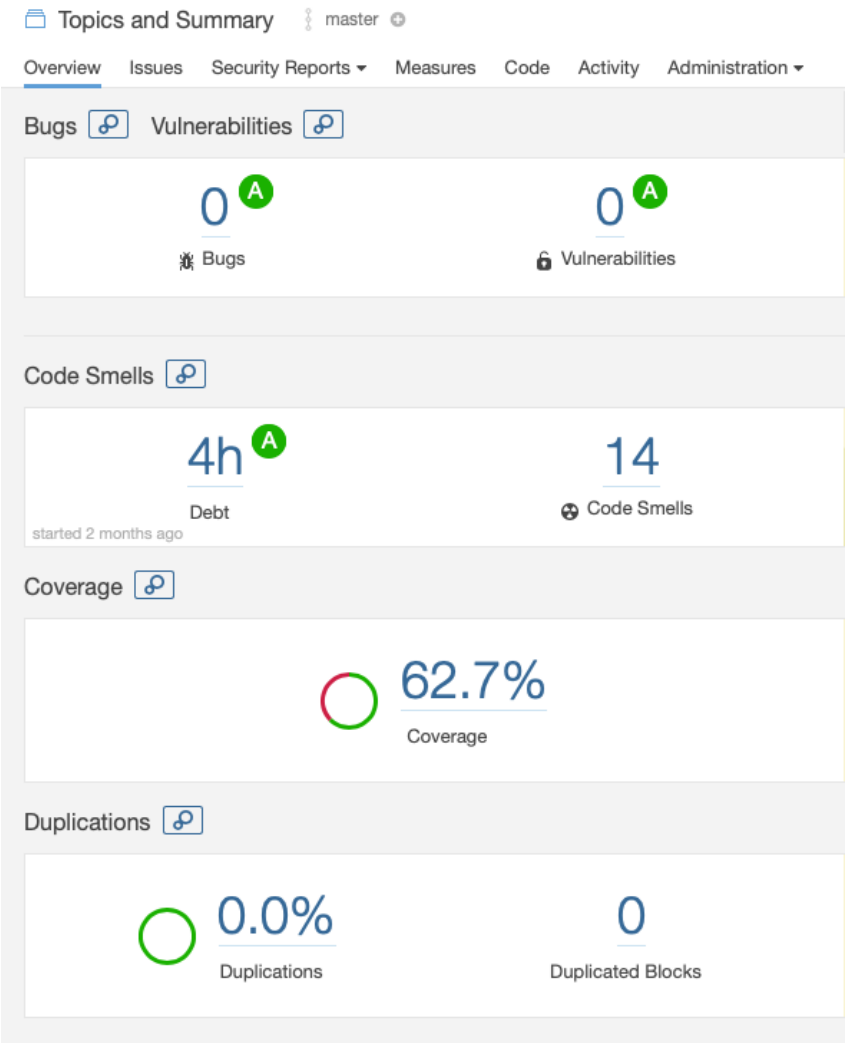



*Figura 8.1: Resultados análisis estático de código (Topics and Summary)*

### 8.2.2 Subsistema Web backend

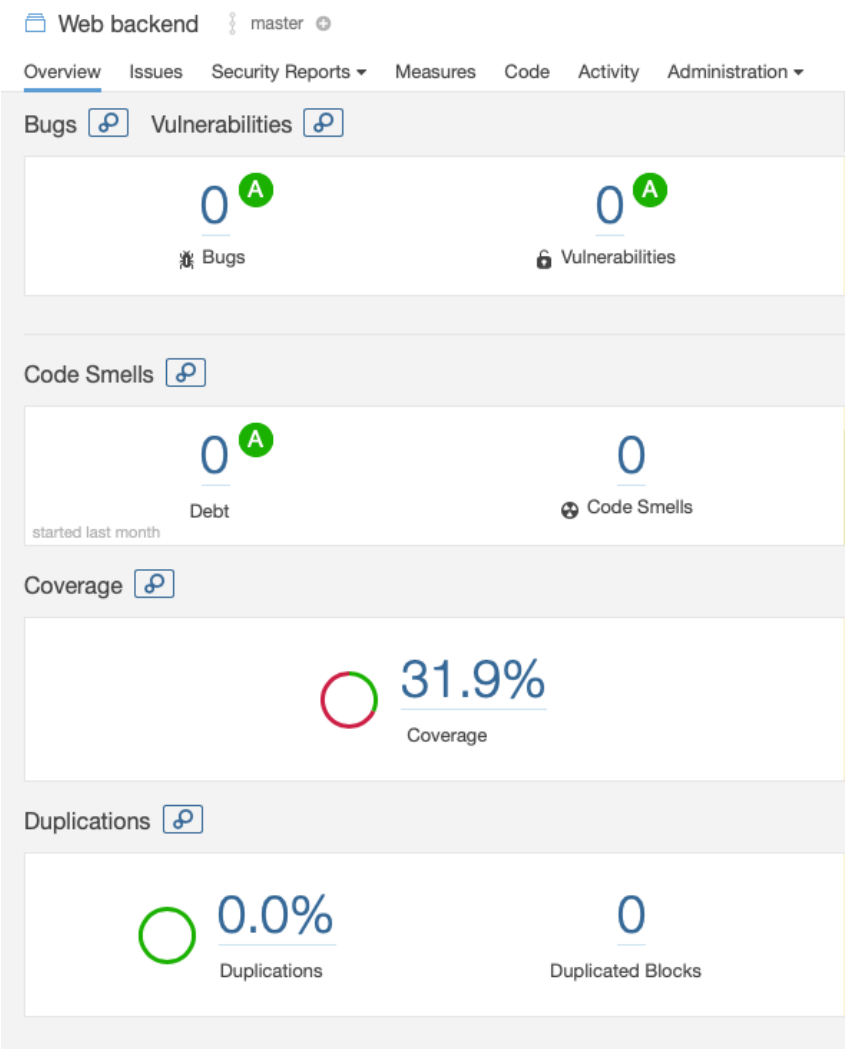



*Figura 8.2: Resultados análisis estático de código (Web backend)*

## 8.3 Pruebas de Usabilidad

A continuación, se muestran los resultados de las pruebas de usabilidad.

#### 8.3.1.1 Usuario 1

##### 8.3.1.1.1 Cuestionario de Evaluación

###### 8.3.1.1.1.1 Preguntas de carácter general

| **¿Utiliza un ordenador, un dispositivo móvil o una tablet frecuentemente?** |
|---|
| 1. Todos los días |
| **¿Qué tipo de actividades realiza con el ordenador / dispositivo móvil / tablet?** |
| 1. Lo uso básicamente para ocio |
| **¿Ha usado alguna vez software como el de esta prueba?** |
| 1. No, nunca |
| **¿Qué busca Vd. Principalmente en un programa?** |
| 1. Que sea intuitivo |
| **¿Qué edad tiene?** |
| 14 |
| **¿Cuál es su profesión?** |
| Estudiante |
| **¿Cómo definiría su nivel de inglés?** |
| 1. Medio |

###### 8.3.1.1.1.2 Problemas o dificultades en las actividades guiadas

El único problema que ha tenido este usuario ha sido con la primera tarea, dado que no le quedaba clara la finalidad de la misma, pero después de explicársela de nuevo, ha podido realizarla correctamente.

###### 8.3.1.1.1.3 Preguntas Cortas sobre la Aplicación y Observaciones

| **Facilidad de Uso** | **Siempre** | **Frecuentemente** | **Ocasionalmente** | **Nunca** |
|---|---|---|---|---|
| *¿Sabe donde está dentro de la aplicación?* | x | | | |
| *¿Existe ayuda para las funciones en caso de que tenga dudas?* | | | x | |
| *¿Le resulta sencillo el uso de la aplicación?* | x | | | |

| Funcionalidad | Siempre | Frecuentemente | Ocasionalmente | Nunca |
|---|---|---|---|---|
| *¿Funciona cada tarea como Vd. Espera?* | x | | | |
| *¿Es capaz de identificar la temática de los topics?* | x | | | |
| *¿Los documentos más representativos de cada topic son coherentes con la temática del mismo?* | | x | | |
| *¿El topic con mayor probabilidad de estar relacionado con el texto que ha introducido es coherente, respecto a la temática del texto?* | x | | | |
| *¿El resumen del texto que ha introducido es bueno (es representativo de dicho texto)?* | x | | | |
| *¿El tiempo de respuesta de la aplicación es muy grande?* | | | x | |
| **Calidad del Interfaz** | | | | |
| **Aspectos gráficos** | **Muy Adecuado** | **Adecuado** | **Poco Adecuado** | **Nada Adecuado** |
| *El tipo y tamaño de letra es* | x | | | |
| *Los iconos e imágenes usados son* | x | | | |
| *Los colores empleados son* | | x | | |
| **Diseño de la Interfaz** | | **Sí** | **No** | **A veces** |
| *¿Le resulta fácil de usar?* | | x | | |
| *¿El diseño de las pantallas es claro y atractivo?* | | x | | |
| *¿Cree que el programa está bien estructurado?* | | x | | |
| **Observaciones** | | | | |
| Al principio es difícil darse cuenta de que el contenido solicitado ha aparecido debajo de la sección correspondiente. Sería conveniente mostrar algún tipo de aviso de que ha aparecido la "tarjeta" con los resultados debajo de su sección (estas "tarjetas" son aquellas en las que aparecen, por ejemplo, los documentos más representativos de un topic, o el resumen del texto introducido por el usuario). | | | | |

#### 8.3.1.1.2 Cuestionario para el Responsable de las Pruebas

| Aspecto Observado | Notas |
|---|---|
| *Tiempo en realizar cada tarea* | Normal en todas las tareas, salvo en la tarea de consultar los topics en formato texto, dado que el usuario no tenía claro qué se le estaba pidiendo. |
| *Errores cometidos* | 1 (en la tarea de consultar los topics en formato texto) |

#### 8.3.1.2 Usuario 2

##### 8.3.1.2.1 Cuestionario de Evaluación

###### 8.3.1.2.1.1 Preguntas de carácter general

| **¿Utiliza un ordenador, un dispositivo móvil o una tablet frecuentemente?** |
|---|
| 2. Varias veces a la semana |
| **¿Qué tipo de actividades realiza con el ordenador / dispositivo móvil / tablet?** |
| 1. Lo uso básicamente para ocio |
| **¿Ha usado alguna vez software como el de esta prueba?** |
| 1. No, nunca |
| **¿Qué busca Vd. Principalmente en un programa?** |
| 1. Que tenga todas las funciones necesarias |
| **¿Qué edad tiene?** |
| 46 |
| **¿Cuál es su profesión?** |
| Ama de casa |
| **¿Cómo definiría su nivel de inglés?** |
| 1. Bajo |

###### 8.3.1.2.1.2 Problemas o dificultades en las actividades guiadas

Este usuario ha tenido problemas con la primera tarea, dado que no le quedaba clara la finalidad de la misma, pero después de explicársela de nuevo, ha podido realizarla correctamente.

También ha tenido problemas con los sliders de la configuración, ya que en primera instancia no sabía como utilizarlos.

###### 8.3.1.2.1.3 Preguntas Cortas sobre la Aplicación y Observaciones

| **Facilidad de Uso** | **Siempre** | **Frecuentemente** | **Ocasionalmente** | **Nunca** |
|---|---|---|---|---|
| *¿Sabe donde está dentro de la aplicación?* | x | | | |
| *¿Existe ayuda para las funciones en caso de que tenga dudas?* | | | x | |
| *¿Le resulta sencillo el uso de la aplicación?* | | x | | |
| **Funcionalidad** | **Siempre** | **Frecuentemente** | **Ocasionalmente** | **Nunca** |
| *¿Funciona cada tarea como Vd. Espera?* | x | | | |

| | | | | |
|---|---|---|---|---|
| *¿Es capaz de identificar la temática de los topics?* | | x | | |
| *¿Los documentos más representativos de cada topic son coherentes con la temática del mismo?* | x | | | |
| *¿El topic con mayor probabilidad de estar relacionado con el texto que ha introducido es coherente, respecto a la temática del texto?* | x | | | |
| *¿El resumen del texto que ha introducido es bueno (es representativo de dicho texto)?* | x | | | |
| *¿El tiempo de respuesta de la aplicación es muy grande?* | | | x | |
| **Calidad del Interfaz** | | | | |
| **Aspectos gráficos** | **Muy Adecuado** | **Adecuado** | **Poco Adecuado** | **Nada Adecuado** |
| *El tipo y tamaño de letra es* | | x | | |
| *Los iconos e imágenes usados son* | x | | | |
| *Los colores empleados son* | x | | | |
| **Diseño de la Interfaz** | | **Sí** | **No** | **A veces** |
| *¿Le resulta fácil de usar?* | | x | | |
| *¿El diseño de las pantallas es claro y atractivo?* | | x | | |
| *¿Cree que el programa está bien estructurado?* | | x | | |
| **Observaciones** | | | | |
| • Que la página se pueda utilizar con textos en español.<br>• Mostrar más ayudas que permitan saber al usuario que la información que solicita va a aparecer debajo de las secciones principales. | | | | |

#### 8.3.1.2.2 Cuestionario para el Responsable de las Pruebas

| **Aspecto Observado** | **Notas** |
|---|---|
| *Tiempo en realizar cada tarea* | Tiempo por encima de lo normal en todas las tareas. |
| *Errores cometidos* | 2 (en la tarea de consultar los topics en formato texto, y a la hora de modificar la configuración de los topics) |

### *8.3.1.3 Usuario 3*

#### 8.3.1.3.1 Cuestionario de Evaluación

##### *8.3.1.3.1.1 Preguntas de carácter general*

| **¿Utiliza un ordenador, un dispositivo móvil o una tablet frecuentemente?** |
|---|
| 1. Todos los días |

| **¿Qué tipo de actividades realiza con el ordenador / dispositivo móvil / tablet?** |
|---|
| 1. Es parte de mi trabajo o profesión |
| **¿Ha usado alguna vez software como el de esta prueba?** |
| 1. No, nunca |
| **¿Qué busca Vd. Principalmente en un programa?** |
| 1. Que tenga todas las funciones necesarias |
| **¿Qué edad tiene?** |
| 22 |
| **¿Cuál es su profesión?** |
| Estudiante |
| **¿Cómo definiría su nivel de inglés?** |
| 1. Alto |

##### *8.3.1.3.1.2 Problemas o dificultades en las actividades guiadas*

Ninguno.

##### *8.3.1.3.1.3 Preguntas Cortas sobre la Aplicación y Observaciones*

| **Facilidad de Uso** | **Siempre** | **Frecuentemente** | **Ocasionalmente** | **Nunca** |
|---|---|---|---|---|
| *¿Sabe donde está dentro de la aplicación?* | x | | | |
| *¿Existe ayuda para las funciones en caso de que tenga dudas?* | | x | | |
| *¿Le resulta sencillo el uso de la aplicación?* | x | | | |
| **Funcionalidad** | **Siempre** | **Frecuentemente** | **Ocasionalmente** | **Nunca** |
| *¿Funciona cada tarea como Vd. Espera?* | x | | | |
| *¿Es capaz de identificar la temática de los topics?* | | x | | |
| *¿Los documentos más representativos de cada topic son coherentes con la temática del mismo?* | x | | | |
| *¿El topic con mayor probabilidad de estar relacionado con el texto que ha introducido es coherente, respecto a la temática del texto?* | x | | | |
| *¿El resumen del texto que ha introducido es bueno (es representativo de dicho texto)?* | | x | | |
| *¿El tiempo de respuesta de la* | | x | | |

| | | | | |
|---|---|---|---|---|
| *aplicación es muy grande?* | | | | |
| **Calidad del Interfaz** | | | | |
| **Aspectos gráficos** | **Muy Adecuado** | **Adecuado** | **Poco Adecuado** | **Nada Adecuado** |
| *El tipo y tamaño de letra es* | x | | | |
| *Los iconos e imágenes usados son* | x | | | |
| *Los colores empleados son* | | x | | |
| **Diseño de la Interfaz** | | **Sí** | **No** | **A veces** |
| *¿Le resulta fácil de usar?* | | x | | |
| *¿El diseño de las pantallas es claro y atractivo?* | | x | | |
| *¿Cree que el programa está bien estructurado?* | | x | | |
| **Observaciones** | | | | |
| Sería conveniente avisar al usuario de que ha aparecido nuevo contenido al final de cada sección después de realizar ciertas acciones, como solicitar los documentos más representativos de un topic. | | | | |

#### 8.3.1.3.2 Cuestionario para el Responsable de las Pruebas

| Aspecto Observado | Notas |
|---|---|
| *Tiempo en realizar cada tarea* | Bajo |
| *Errores cometidos* | 0 |

### *8.3.1.4 Resultados finales y conclusiones*

#### 8.3.1.4.1 Evaluación de aspectos de usabilidad

- **Facilidad de aprendizaje:** En general, los usuarios tienen pocos problemas al utilizar el sistema. El tiempo que tardan en realizar las tareas es bajo, salvo la primera vez que realizan cada tarea, que tardan algo más, dado que tienen que analizar la interfaz para encontrar la manera de realizarla, pero nada fuera de lo normal. Después de haber realizado cada tarea una vez, se pueden desenvolver sin problemas por todo el sistema.
- **Errores:** El número de errores es bajo. El error más recurrente se da en la primera tarea, que es la de consultar los topics en formato texto. Este error no es preocupante, y se debe a varias cuestiones:
    - Es la primera tarea, por lo que es la primera toma de contacto de los usuarios con el sistema
    - La formulación de la tarea no es del todo clara, y suele confundir a los usuarios.
    - El formato nube de palabras, aunque muestra imágenes, son imágenes de palabras, por lo que esto también causa algo de confusión.
    - El resultado de esta tarea aparece directamente al cargar la página. Los usuarios se esperan que el resultado de esta tarea aparezca en un lugar distinto, y no consideran que lo que muestra la página inicialmente pueda ser el resultado de la tarea.

- **Satisfacción del usuario:** Los usuarios quedan satisfechos con el sistema. La primera toma de contacto, como suele ser habitual, les cuesta más, pero después aseguran que serían capaces de utilizarlo sin problemas. Lo más solicitado es la inclusión de más ayudas, sobretodo para entender como funciona el sistema la primera vez que lo utilizan, dado que se les ha dejado solos frente al mismo, explicándoles solamente la idea que hay detrás del sistema, pero no el funcionamiento de la interfaz de usuario. Una opinión muy recurrente es la inclusión de algún tipo de notificación que indique al usuario que ha aparecido debajo de la sección una "tarjeta" con la información solicitada. Por último, los usuarios consideran que los tiempos de respuesta algo elevados en ciertas funcionalidades de la aplicación podrían llegar a suponer un problema en caso de que tuvieran que utilizar el sistema de forma frecuente.

#### 8.3.1.4.2 Conclusiones de las pruebas

A continuación, se exponen una serie de posibles cambios a realizar en el sistema, en base a los resultados de estas pruebas, así como una opinión respecto a si dicho cambio es factible, que implicaría llevarlo a cabo y/o como podría realizarse:

- **Aumentar el número de mensajes de ayuda:** Dado que los problemas suelen ser con la primera toma de contacto con el sistema, se podría añadir en la parte superior de la página un pequeño icono de ayuda que al hacer click en él muestre una imagen resumiendo como funciona la aplicación, e indicando en qué partes de la misma se realizan cada una de las funcionalidades. Aquí sería importante recalcar que gran parte de la información solicitada por el usuario aparece de forma dinámica debajo de la sección correspondiente.
- **Disminuir el tiempo que tardan los resúmenes en generarse:** Los resúmenes son los que hacen que se incremente el tiempo de respuesta en gran parte de las funcionalidades del sistema, por lo que, si se reduce el tiempo de la generación de resúmenes, se reduce el tiempo de respuesta en la mayor parte del sistema. Para conseguir esto, se podría tratar de buscar otro algoritmo que genere resúmenes en menor tiempo que TextRank, aunque la opción más factible es generar los resúmenes de los textos del dataset en batch, es decir, calcular dichos resúmenes previamente y almacenarlos en el backend, para así no tener que generarlos en el momento, tal y como se hace ahora. Esto tiene un inconveniente, y es que se incrementa la cantidad de espacio de almacenamiento necesaria en el servidor del backend, pero teniendo en cuenta que el tiempo de respuesta se reduciría drásticamente, y que el espacio necesario para almacenar los resúmenes sería bajo (con la configuración actual se almacenarían 2 frases por cada documento de texto, y el dataset actual consta de 20.000 documentos) considero que es un cambio que merece la pena llevar a cabo.
- **Permitir trabajar con datasets en español:** Este cambio no es sencillo y llevaría tiempo, dado que implicaría añadir otro módulo de preprocesado de texto en español, así como buscar vectores de palabras en español preentrenados utilizando word embedding, o generarlos utilizando una implementación de Glove o de Word2Vec. Aún así, a pesar de no ser sencillo, es un cambio interesante, y sería necesario si el sistema se quisiera utilizar con textos en español.

- **Añadir un scroll automático que, posteriormente a que el usuario solicite cierta información, le desplace al contenido que acaba de aparecer debajo de la sección:** De esta forma, se facilita al usuario encontrar dicha información.

#### 8.3.1.4.3 Cambios producidos en la interfaz a raíz de estas pruebas

De todos los posibles cambios expuestos en la sección anterior, se ha llevado a cabo el relativo al **scroll automático**.

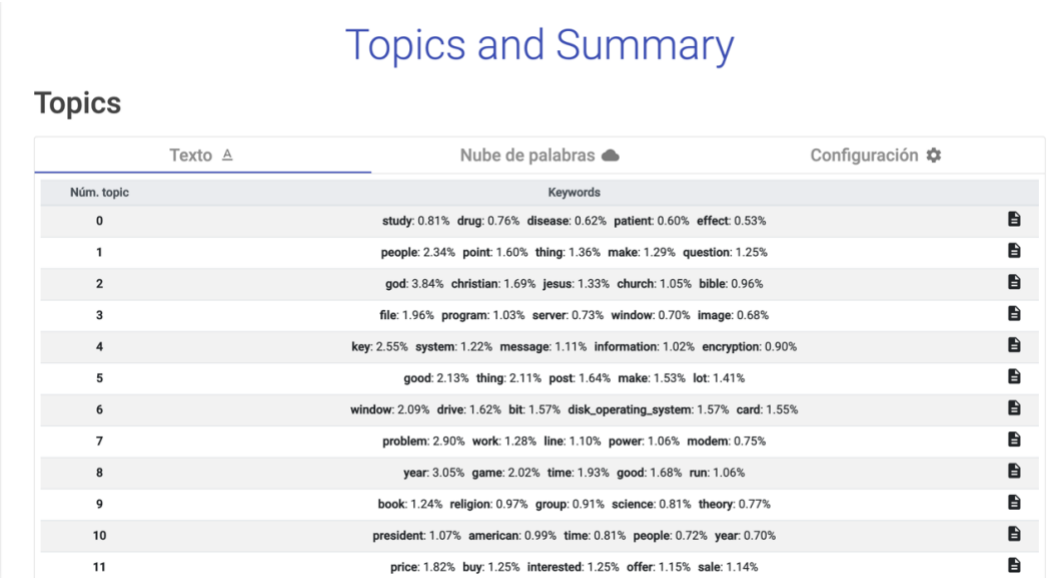
El resto de cambios se dejan como posibles cambios futuros en la sección  Ampliaciones.

## 8.4 Pruebas de adaptabilidad

A continuación, se muestran capturas de pantalla con los resultados de las pruebas de adaptabilidad, con las siguientes resoluciones:

- 1440 x 768 (Opción Latptop L de Google Chrome)
- 1024 x 768 (Opción Laptop de Google Chrome)
- 700 x 768
- 320 x 768 (Opción Mobile S de Google Chrome)

### 8.4.1 Resolución 1440 x 768




| Núm. topic | Keywords |
|---|---|
| 0 | study: 0.81% drug: 0.76% disease: 0.62% patient: 0.60% effect: 0.53% |
| 1 | people: 2.34% point: 1.60% thing: 1.36% make: 1.29% question: 1.25% |
| 2 | god: 3.84% christian: 1.69% jesus: 1.33% church: 1.05% bible: 0.96% |
| 3 | file: 1.96% program: 1.03% server: 0.73% window: 0.70% image: 0.68% |
| 4 | key: 2.55% system: 1.22% message: 1.11% information: 1.02% encryption: 0.90% |
| 5 | good: 2.13% thing: 2.11% post: 1.64% make: 1.53% lot: 1.41% |
| 6 | window: 2.09% drive: 1.62% bit: 1.57% disk_operating_system: 1.57% card: 1.55% |
| 7 | problem: 2.90% work: 1.28% line: 1.10% power: 1.06% modem: 0.75% |
| 8 | year: 3.05% game: 2.02% time: 1.93% good: 1.68% run: 1.06% |
| 9 | book: 1.24% religion: 0.97% group: 0.91% science: 0.81% theory: 0.77% |
| 10 | president: 1.07% american: 0.99% time: 0.81% people: 0.72% year: 0.70% |
| 11 | price: 1.82% buy: 1.25% interested: 1.25% offer: 1.15% sale: 1.14% |

*Figura 8.3: Resolución 1440x768 (topics – formato texto)*

*Figura 8.4: Resolución 1440x768 (topics – formato nube de palabras)*

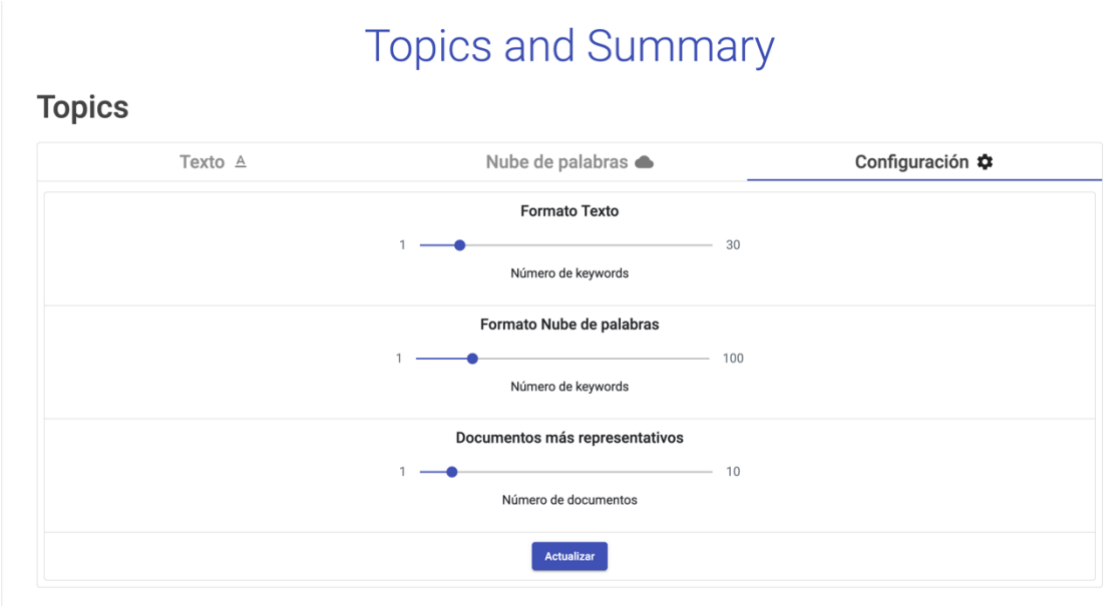



*Figura 8.5: Resolución 1440x768 (topics – configuración)*

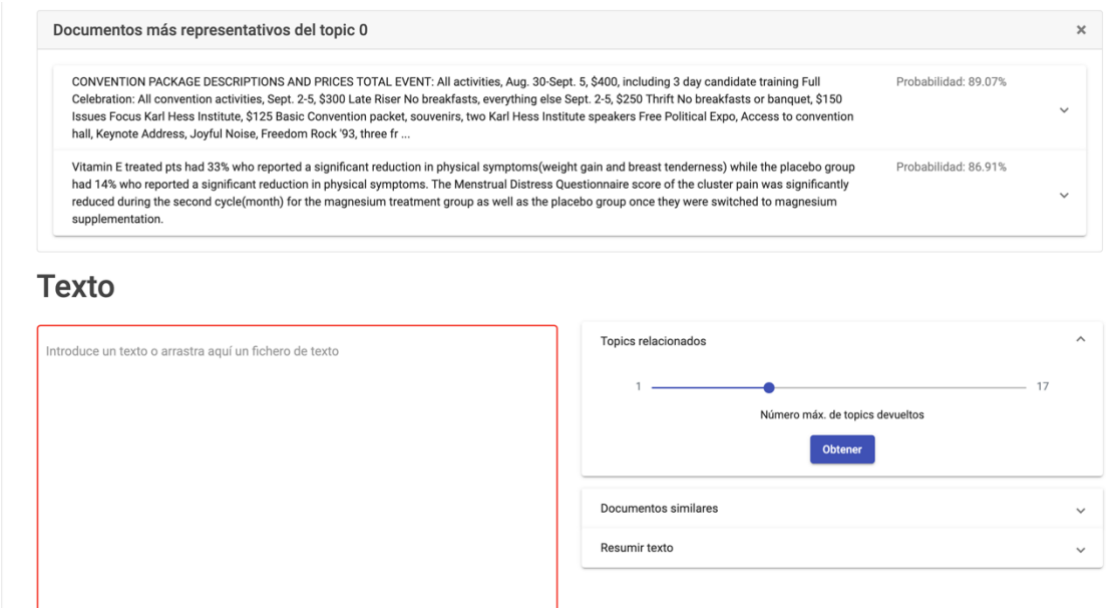



*Figura 8.6: Resolución 1440x768 (documentos de topic y sección texto)*

*Figura 8.7: Resolución 1024x768 (texto – prob. topics, documentos similares y resumen)*

### 8.4.2 Resolución 1024 x 768

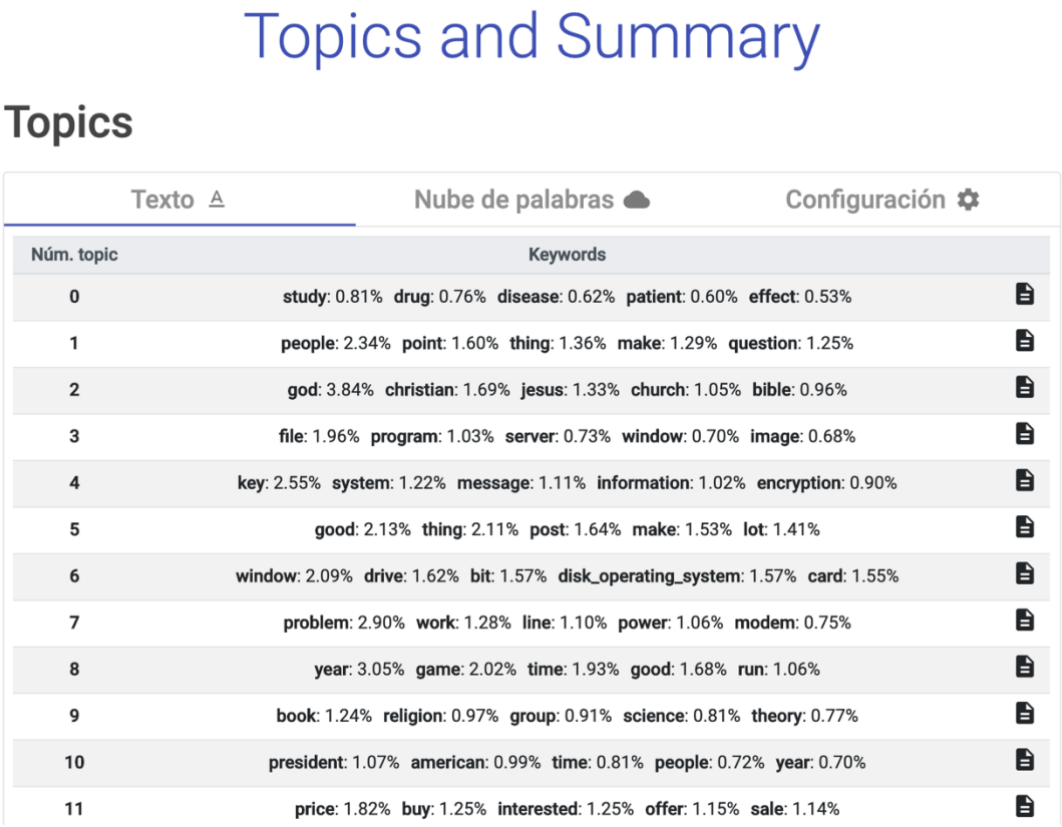



*Figura 8.8: Resolución 1024x768 (topics – formato texto)*

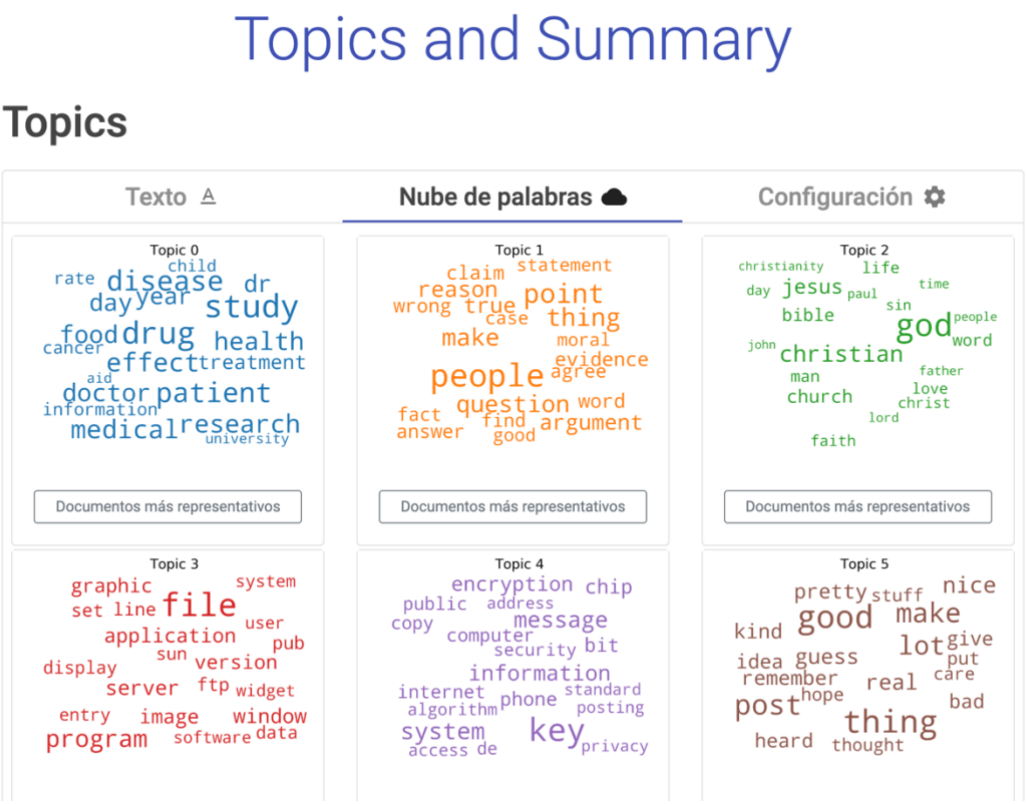



*Figura 8.9: Resolución 1024x768 (topics – formato nube de palabras)*

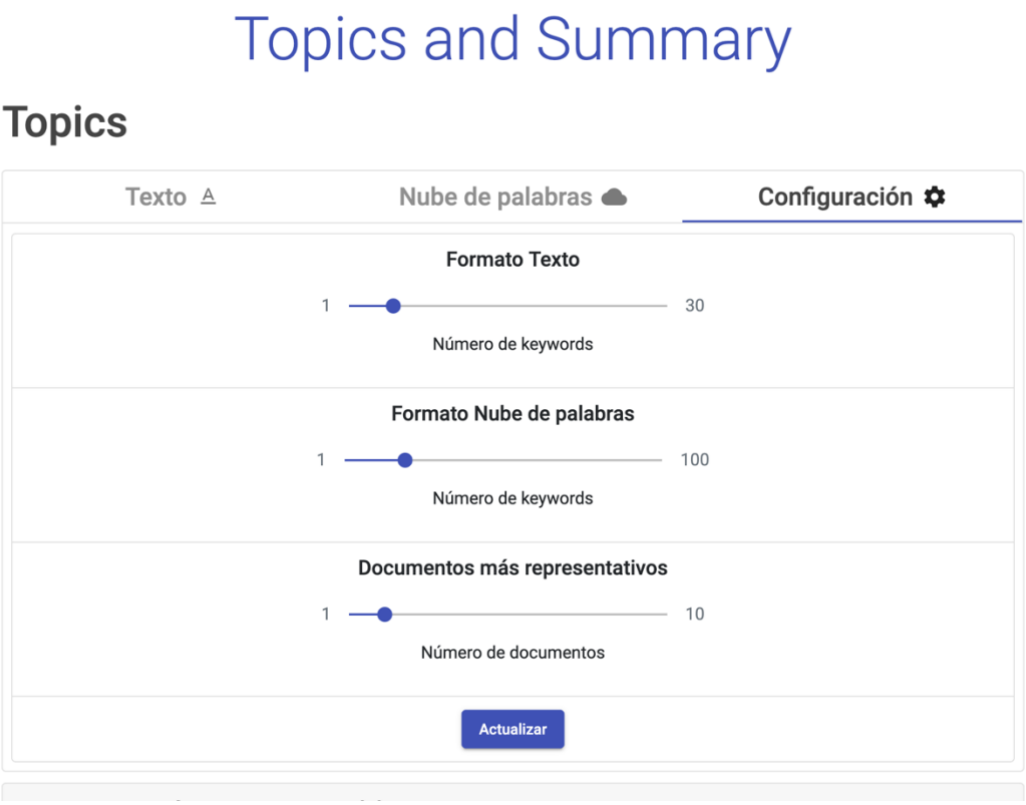



*Figura 8.10: Resolución 1024x768 (topics – configuración)*

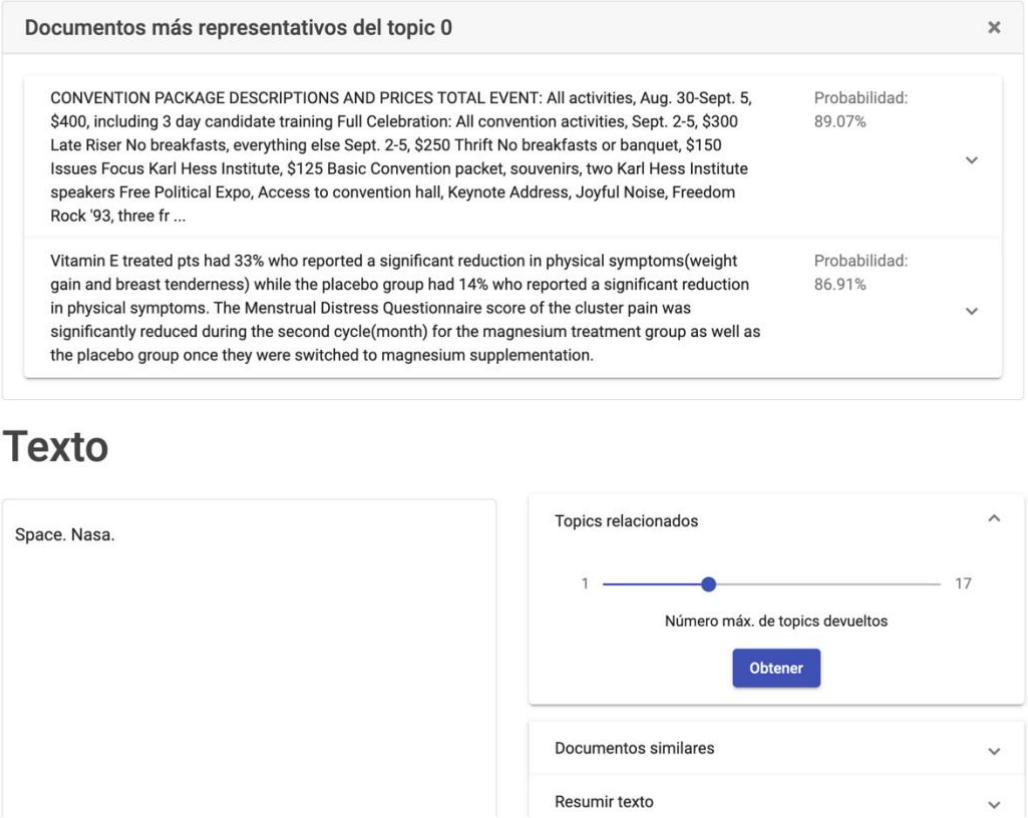



*Figura 8.11: Resolución 1024x768 (documentos de topic y sección texto)*

*Figura 8.12: Resolución 1024x768 (texto – prob. topics, documentos similares y resumen)*

### 8.4.3 Resolución 700 x 768

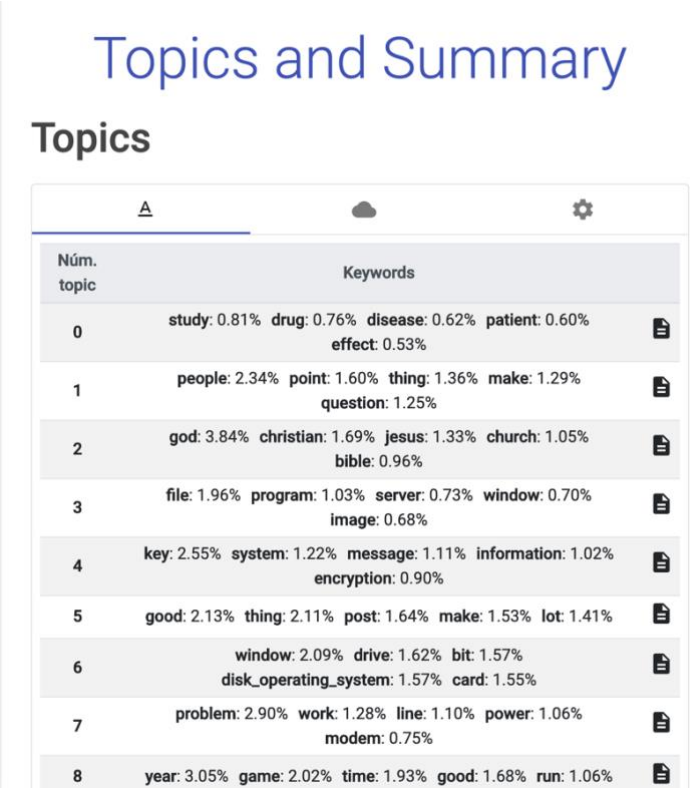



*Figura 8.13: Resolución 700x768 (topics – formato texto)*

*Figura 8.14: Resolución 700x768 (topics – formato nube de palabras)*

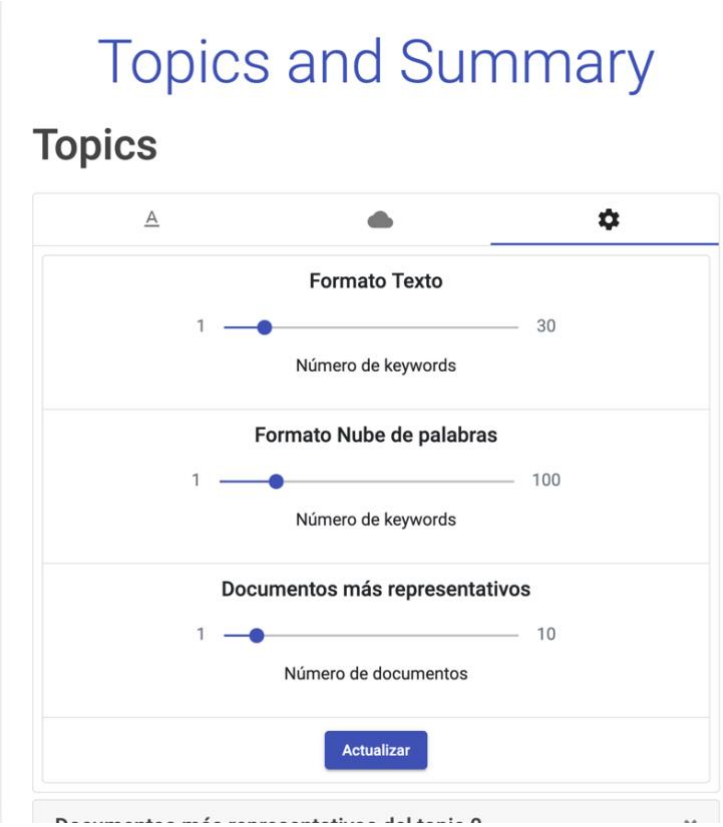



Figura 8.15: Resolución 700x768 (topics – configuración)

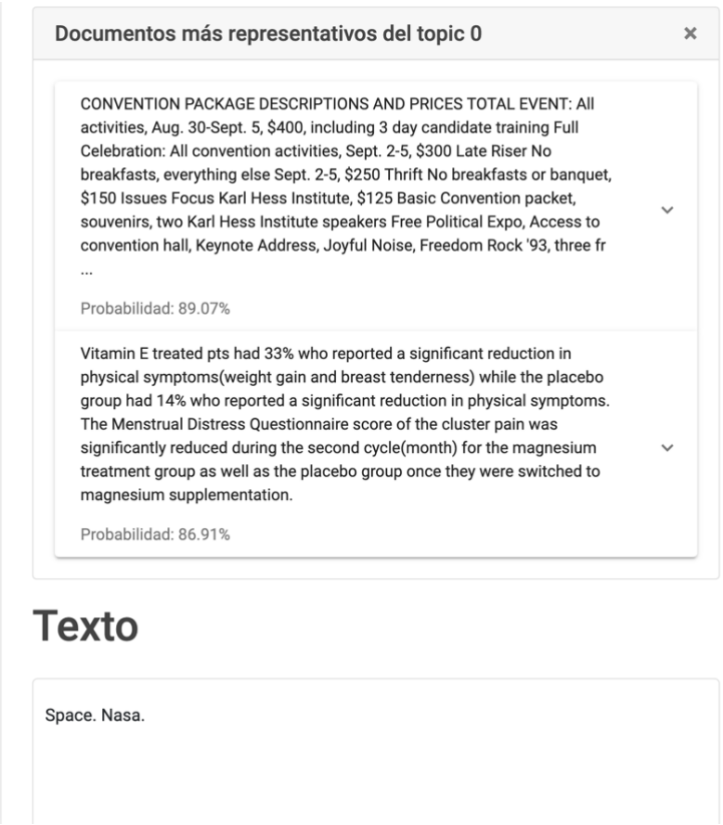



Figura 8.16: Resolución 700x768 (documentos de topic y sección texto)

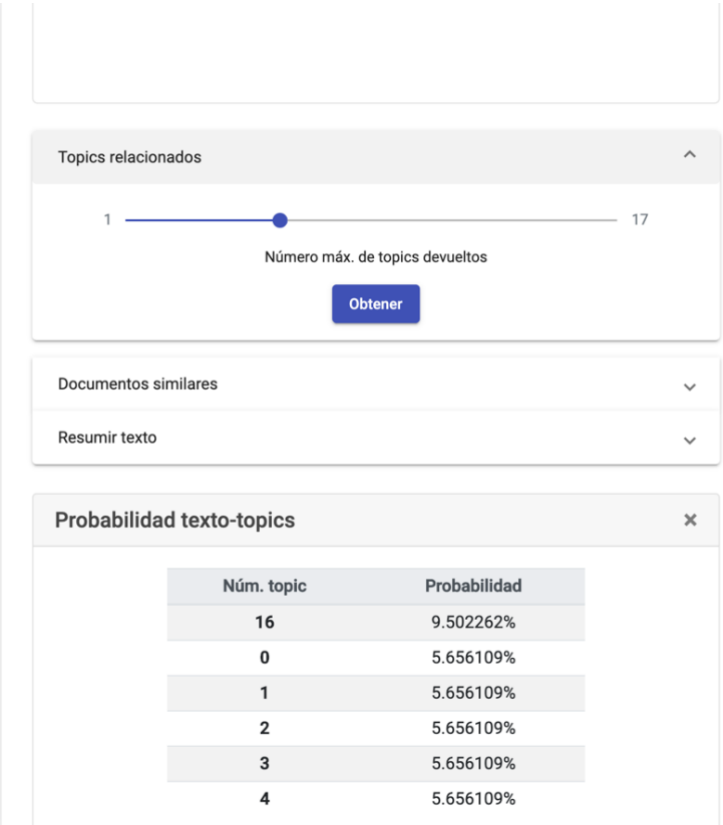



Figura 8.17: Resolución 700x768 (texto – configuración y prob. topics)

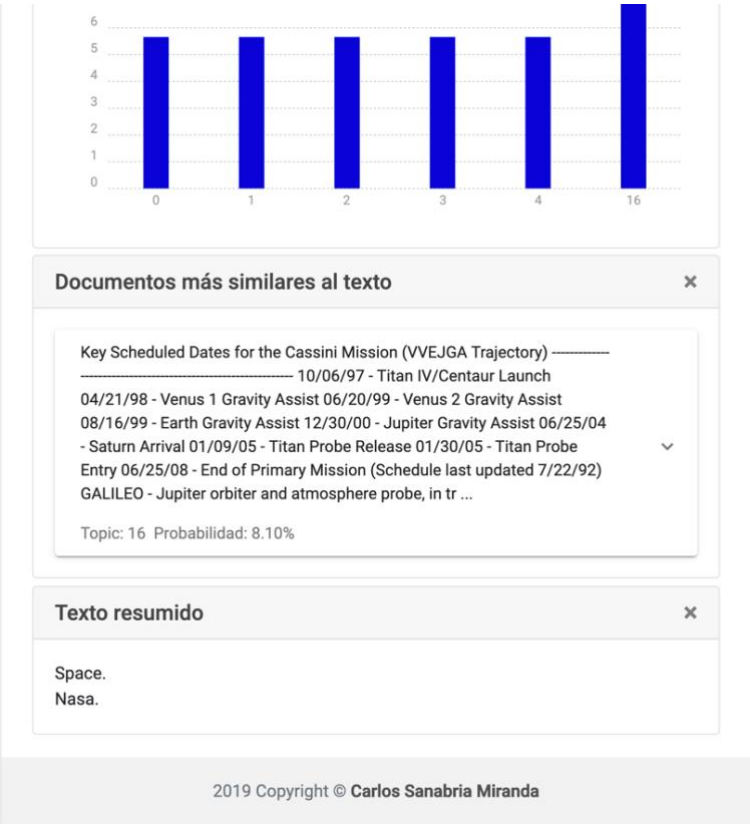



Figura 8.18: Resolución 700x768 (texto – prob. topics, documentos similares y resumen)

### 8.4.4 Resolución 320 x 768

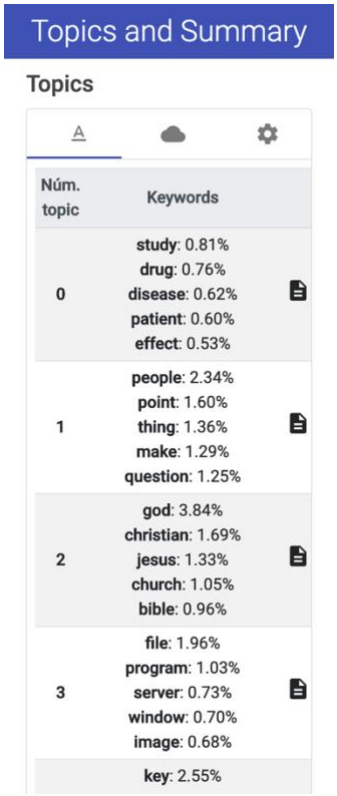



*Figura 8.19: Resolución 320x768 (topics – formato texto)*

*Figura 8.20: Resolución 320x768 (topics – formato nube de palabras)*

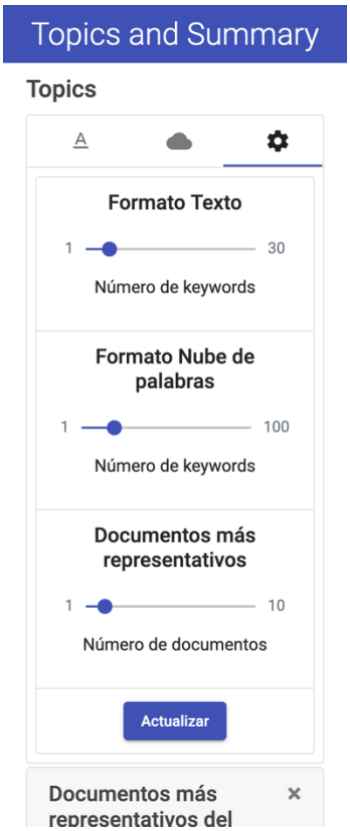



Figura 8.21: Resolución 320x768 (topics – configuración)

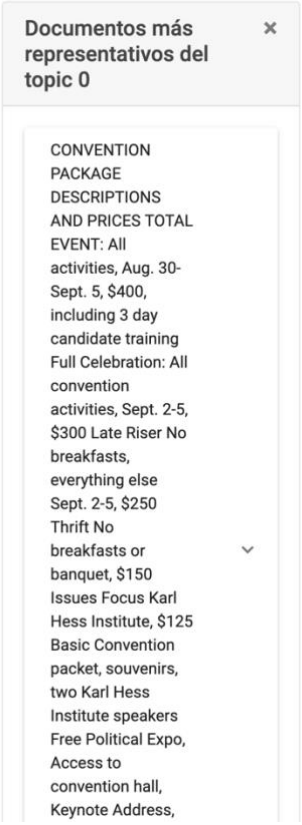



Figura 8.22: Resolución 320x768 (documentos de topic)

Figura 8.23: Resolución 320x768 (sección texto)

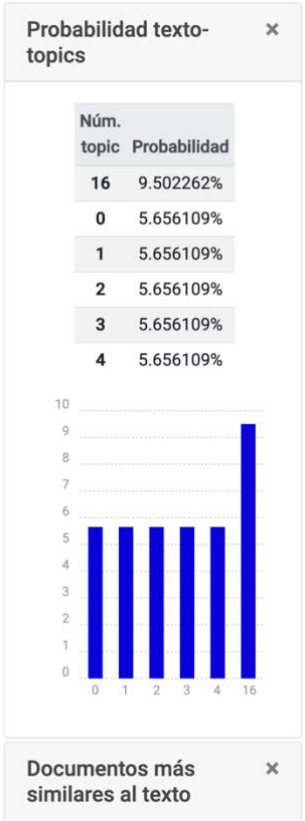



Figura 8.24: Resolución 320x768 (texto – prob. topics)

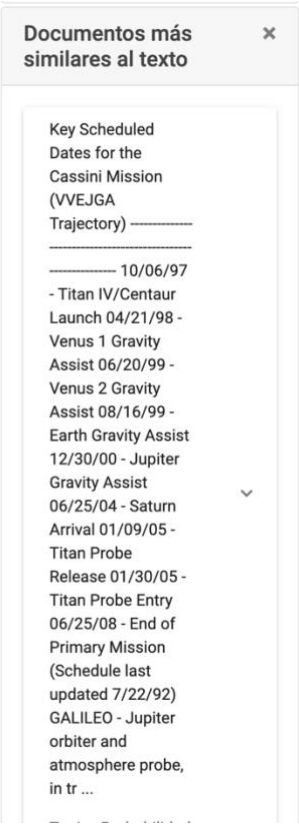



*Figura 8.25: Resolución 700x768 (texto –documentos similares)*

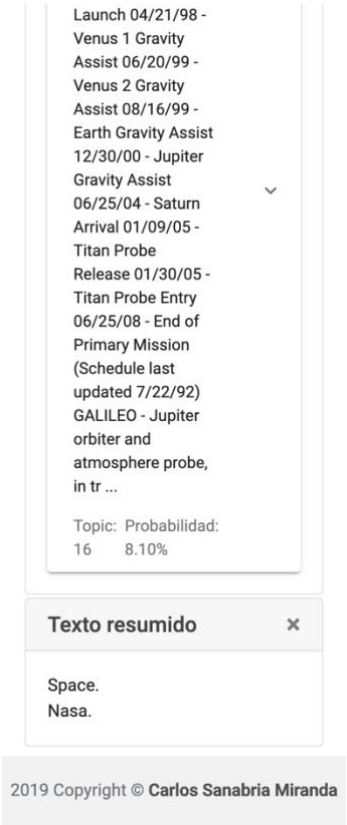



*Figura 8.26: Resolución 700x768 (texto – resumen)*

# Capítulo 9. Manuales del Sistema

## 9.1 Manual de Instalación

Los manuales de instalación se encuentran dentro de la propia documentación de cada subsistema. En ellos, se explican todos los pasos necesarios para configurar un entorno en el que posteriormente ejecutar cada subsistema, incluyendo la instalación de lenguajes, frameworks, y librerías utilizadas.

A continuación, se indican las rutas para acceder a ellos, dentro del Archivo adjunto.

#### *9.1.1.1 Subsistema Topics and Summary*

Este subsistema contiene 2 manuales para su instalación: Uno enfocado en cómo instalarlo para su utilización, y otro enfocado en cómo instalarlo para su desarrollo.

##### 9.1.1.1.1 Manual de instalación para utilizar el subsistema

**Ruta:** topics_and_summary/topics_and_summary/docs/build/html/usage/installation.html

##### 9.1.1.1.2 Manual de instalación para desarrollar el subsistema

**Ruta:**
topics_and_summary/topics_and_summary/docs/build/html/development/installation.html

#### *9.1.1.2 Web backend*

Este subsistema contiene un manual para la instalación de las dependencias necesarias para ejecutarlo.

**Ruta:** web_backend/web_backend/docs/build/html/dependencies-and-files.html

#### *9.1.1.3 Web frontend*

Este subsistema contiene un manual para instalar Node.js, Angular CLI, y las dependencias utilizadas en él.

**Ruta:** web-frontend/documentation/additional-documentation/installation.html

## 9.2 Manual de Ejecución y Despliegue

Los manuales de ejecución y despliegue también se encuentran dentro de la propia documentación de cada subsistema. En ellos, se explican todos los pasos necesarios para ejecutar y desplegar cada uno de los subsistemas.

A continuación, se indican las rutas para acceder a ellos, dentro del Archivo adjunto.

#### 9.2.1.1 Subsistema Topics and Summary

Este subsistema contiene un manual en el que se explican los pasos necesarios para ejecutar una demo que muestra sus funcionalidades, mediante interacción a través de la consola. La demo puede ser ejecutada de forma normal, o bien utilizando Docker o Vagrant. Además, el manual también explica paso a paso cada una de las funcionalidades mostradas en la demo.

**Ruta:** topics_and_summary/topics_and_summary/docs/build/html/usage/quickstart.html

#### 9.2.1.2 Web backend

Este subsistema contiene un manual en el que se explica cómo ejecutar el backend en desarrollo desde la línea de comandos, cómo instalarlo a mano en un entorno de producción, cómo generar una imagen Docker que ejecute el backend, y cómo desplegar dicha imagen Docker en Heroku.

**Ruta:** web_backend/web_backend/docs/build/html/execution-and-deployment.html

#### 9.2.1.3 Web frontend

Este subsistema contiene un manual en el que se explica cómo ejecutar el frontend en desarrollo, cómo generar una imagen Docker que ejecute el frontend, y cómo desplegar dicha imagen Docker en Heroku.

**Ruta:**
web-frontend/documentation/additional-documentation/execution-and-deployment.html

## 9.3 Manual del Programador

Los manuales de programador también se encuentran dentro de la propia documentación de cada subsistema. En ellos se explican, entre otras cosas, la guía de estilos y las librerías más importantes utilizadas, la estructura de directorios y los documentos más relevantes, aspectos importantes del código fuente, y detalles que faciliten continuar con su desarrollo.

A continuación, se indican las rutas para acceder a ellos, dentro del Archivo adjunto.

#### 9.3.1.1 Subsistema Topics and Summary

Además de lo mencionado previamente, el manual del programador de este subsistema explica: los conceptos fundamentales de la extracción de topics y de los modelos de generación de resúmenes, y cómo realizar un análisis estático del código fuente usando un servidor SonarQube local.

**Ruta:** topics_and_summary/topics_and_summary/docs/build/html/development/api.html

#### 9.3.1.2 Web backend

Al igual que el manual del programador del subsistema anterior, este también explica cómo realizar un análisis estático del código fuente usando un servidor SonarQube local.

**Ruta:** web_backend/web_backend/docs/build/html/development/api.html

#### 9.3.1.3 Web frontend

**Ruta:** web-frontend/documentation/additional-documentation/development/development:-api.html

## 9.4 Documentación adicional

La documentación de cada subsistema, además de los manuales mencionados en las secciones anteriores, también contiene información relativa a cómo continuar con el desarrollo de dicha documentación, así como posibles mejoras futuras que se pueden realizar en cada subsistema, tanto a nivel de código, como a nivel de documentación.

A continuación, se indican las rutas para acceder a dicha documentación, dentro del Archivo adjunto.

#### *9.4.1.1 Subsistema Topics and Summary*

##### 9.4.1.1.1 Desarrollo de la documentación

**Ruta:** topics_and_summary/topics_and_summary/docs/build/html/development/documentation.html

##### 9.4.1.1.2 Mejoras futuras

**Ruta:** topics_and_summary/topics_and_summary/docs/build/html/development/future-improvements.html

#### *9.4.1.2 Web backend*

##### 9.4.1.2.1 Desarrollo de la documentación

**Ruta:** web_backend/web_backend/docs/build/html/development/documentation.html

##### 9.4.1.2.2 Mejoras futuras

**Ruta:** web_backend/web_backend/docs/build/html/development/future-improvements.html

#### *9.4.1.3 Web frontend*

##### 9.4.1.3.1 Desarrollo de la documentación

**Ruta:** web-frontend/documentation/additional-documentation/development/development:-documentation.html

##### 9.4.1.3.2 Mejoras futuras

**Ruta:** web-frontend/documentation/additional-documentation/development/future-improvements.html

# Capítulo 10. Conclusiones y Ampliaciones

## 10.1 Conclusiones

El sistema implementado **cumple con los objetivos** propuestos inicialmente en las secciones *Resumen de la Motivación, Objetivos y Alcance del Proyecto* y *Objetivos del Proyecto*.

El **modelo de extracción de topics** ofrece unos resultados bastante buenos, y el **modelo de generación de resúmenes**, a falta de realizar algún tipo de evaluación de los resúmenes generados, parece ofrecer también buenos resultados, aunque el tiempo que tarda en resumir textos muy grandes es en ocasiones elevado, teniendo en cuenta lo acostumbrados que estamos a tener los resultados prácticamente en tiempo real.

En cuanto a la **API REST** implementada en el **backend**, se ha seguido de la forma más fiel posible la filosofía orientada a recursos, aún cuando en el caso de este sistema no se dispone de recursos como tal, sino de entidades más abstractas, como topics o texto.

En cuanto a la **interfaz de usuario** implementada en el **frontend**, cumple totalmente las expectativas iniciales, siguiendo fielmente la interfaz diseñada en la sección *Diseño de la Interfaz*. De entre las pocas cosas que se han implementado de manera distinta en la interfaz final con respecto al prototipo, destacar la ventana de configuración de los topics, que en lugar de implementarse como una ventana modal se ha implementado como una pestaña más.

Los **lenguajes, frameworks y librerías** utilizados, también han cumplido las expectativas, facilitando mucho la implementación de este sistema. La **arquitectura** en 3 subsistemas ha sido todo un acierto, permitiendo separar claramente las responsabilidades, y permitiendo desplegar por separado el backend y el frontend.

En cuanto a las **herramientas** utilizadas para el desarrollo, también han supuesto un acierto. Los IDEs escogidos han facilitado mucho el desarrollo de los subsistemas, y la utilización de Docker ha simplificado el despliegue, permitiendo subir directamente las imágenes Docker a Heroku. Por otro lado, Google Chrome ha permitido depurar fácilmente el frontend, así como realizar las pruebas de adaptabilidad de manera rápida y sencilla.

Como **conclusión**, me gustaría destacar lo mucho que ha supuesto este proyecto en materia de **aprendizaje**:

- Se ha profundizado en el campo del **Procesamiento de Lenguaje Natural**, en específico, en el preprocesado de textos, los modelos de word embedding, la extracción de topics, y la generación de resúmenes de textos.
- Se ha pasado por todas las **fases del desarrollo de software** (análisis, diseño, implementación y pruebas), aplicando diversos patrones de diseño y de arquitectura, y utilizando diversos lenguajes, herramientas y tecnologías nuevas.

## 10.2 Ampliaciones

A continuación, se listan una serie de posibles ampliaciones que se pueden aplicar al sistema implementado.

### 10.2.1 Implementar las funcionalidades del administrador

Esta ampliación consiste en implementar las funcionalidades del administrador del sistema, tomando como base lo descrito en los Requisitos del Sistema y en el .

Se debe diseñar e implementar la interfaz de usuario del administrador, así como implementar su API REST dentro del fichero web_backend/web_backend/apis/admin.py.

### 10.2.2 Probar los modelos de extracción de topics con más datasets

Los modelos de extracción de topics solamente han sido probado con el dataset 20newsgroups. Esta ampliación consistiría en probar los modelos LdaMalletModel y LdaGensimModel con otros datasets (cuyos documentos estén en inglés) y evaluar los resultados.

### 10.2.3 Evaluar los resultados del algoritmo TextRank

Evaluar los resúmenes de texto es una tarea muy complicada, y la mayoría de las veces es necesaria la intervención de un experto humano que de una valoración acerca de la calidad de los resúmenes. Existen algunas métricas para evaluar los resúmenes, como ROUGE [20], pero siguen necesitando de un experto que establezca qué frases resumen mejor el texto.

Un posible enfoque para evaluar los resultados del algoritmo TextRank (o de cualquier otro algoritmo de generación de resúmenes) es el siguiente:

1. Seleccionar unos cuantos documentos del dataset utilizado para generar el modelo de extracción de topics
2. Obtener sus resúmenes utilizando el modelo de generación de resúmenes
3. Utilizar el modelo de extracción de topics para, dados los resúmenes de los documentos, obtener su probabilidad de pertenencia a los topics
4. Comparar el topic al que pertenecen los documentos seleccionados con el topic con mayor probabilidad de sus resúmenes
5. El número de aciertos con respecto al número de documentos seleccionados puede ser utilizado como métrica para evaluar la calidad de los resúmenes, dado que lo

esperado es que el resumen de un documento pertenezca al mismo topic que el propio documento

### 10.2.4 Probar otros algoritmos para la generación de los resúmenes

Esta ampliación consiste en probar otros algoritmos para la generación de resúmenes y compararlos con el algoritmo TextRank. Debería combinarse con la ampliación anterior, para así tener una forma más objetiva de evaluar y comparar los resúmenes.

Una visión general de las distintas técnicas para la generación de resúmenes de texto se puede encontrar en [21].

### 10.2.5 Aumentar el número de mensajes de ayuda en la interfaz gráfica

Añadir en la parte superior de la página un pequeño icono de ayuda que al hacer click en él muestre una imagen resumiendo como funciona la aplicación, e indicando en qué partes de la misma se realizan cada una de las funcionalidades. Aquí sería importante recalcar que gran parte de la información solicitada por el usuario aparece de forma dinámica debajo de la sección correspondiente.

### 10.2.6 Generar los resúmenes de los documentos del dataset en batch

Generar los resúmenes de los documentos del dataset en batch (es decir, calcularlos previamente y almacenarlos en el backend), en lugar de generarlos en el momento en el que son solicitados por el usuario, reduciría muchísimo el tiempo de respuesta en gran parte de las funcionalidades del sistema.

Esto tiene un inconveniente, y es que se incrementa la cantidad de espacio de almacenamiento necesaria en el servidor del backend, pero teniendo en cuenta que el tiempo de respuesta se reduciría drásticamente, y que el espacio necesario para almacenar los resúmenes sería bajo, es un cambio que merece la pena llevar a cabo.

### 10.2.7 Ampliar el subsistema Topics and Summary para que permita trabajar con datasets en español

Este cambio no es sencillo y llevaría tiempo, dado que implicaría añadir otro módulo de preprocesado de texto en español, así como buscar vectores de palabras en español preentrenados utilizando word embedding, o generarlos utilizando una implementación de

Glove o de Word2Vec. Aún así, a pesar de no ser sencillo, es un cambio interesante, y sería necesario si el sistema se quisiera utilizar con textos en español.

### 10.2.8 Añadir internacionalización al frontend

Esta ampliación consiste en internacionalizar el subsistema *Web frontend*. Esto implicaría parametrizar todo el texto presente en los ficheros HTML y Typescript que vaya a ser mostrado al usuario a través de la interfaz, y añadir ficheros con las traducciones en distintos idiomas (empezando por español e inglés) de esos valores parametrizados.

Actualmente, Angular no permite añadir internacionalización a los textos presentes en el código Typescript, así como tampoco permite modificar dinámicamente el lenguaje de la aplicación, por lo que, si se quisiera poder visualizar la página tanto en español como en inglés, habría que compilarla y desplegarla en ambos idiomas.

Información sobre cómo realizar la internacionalización se encuentra en [22].

### 10.2.9 Hacer la interfaz web accesible

Esta ampliación consiste en realizar los cambios que sean necesarios en el subsistema *Web frontend* para garantizar la accesibilidad de la interfaz web. Una opción sería seguir la recomendación WCAG 2.1 del W3C, descrita en [23].

### 10.2.10 Otras ampliaciones

En el apartado "*Future improvements*" de la documentación de cada subsistema se encuentran otras posibles ampliaciones, de más bajo nivel. En la sección *Documentación adicional* se indican las rutas de estos apartados, dentro del Archivo adjunto.

# Capítulo 11. Planificación del Proyecto y Presupuesto finales

## 11.1 Planificación Final

El proyecto ha tenido una duración total de **450 horas**. La fecha de **comienzo** ha sido el **1 de febrero de 2019**, y la fecha de **finalización** el **7 de julio de 2019**.

Finalmente, el proyecto se ha dividido en las siguientes 8 etapas:

| Etapa | Duración | Fecha de comienzo | Fecha de fin |
|---|---|---|---|
| **Estudios iniciales** | 25 horas | 01/02/2019 | 09/02/2019 |
| **Documentación inicial** | 5 horas | 09/02/2019 | 10/02/2019 |
| **Planificación y presupuestos iniciales** | 13 horas | 10/02/2019 | 16/02/2019 |
| **Análisis** | 25 horas | 16/02/2019 | 24/02/2019 |
| **Diseño** | 36,5 horas | 24/02/2019 | 09/03/2019 |
| **Implementación** | 258,5 horas | 09/03/2019 | 08/06/2019 |
| **Implementación de las pruebas** | 13 horas | 08/06/2019 | 11/06/2019 |
| **Resto de la documentación** | 74 horas | 12/06/2019 | 07/07/2019 |

Las actividades de la etapa de **despliegue** que se había considerado en la planificación inicial se han incluido dentro de la etapa de implementación, por eso hay una etapa menos.

A continuación, se muestra un gráfico circular que compara la duración de las etapas mencionadas anteriormente:

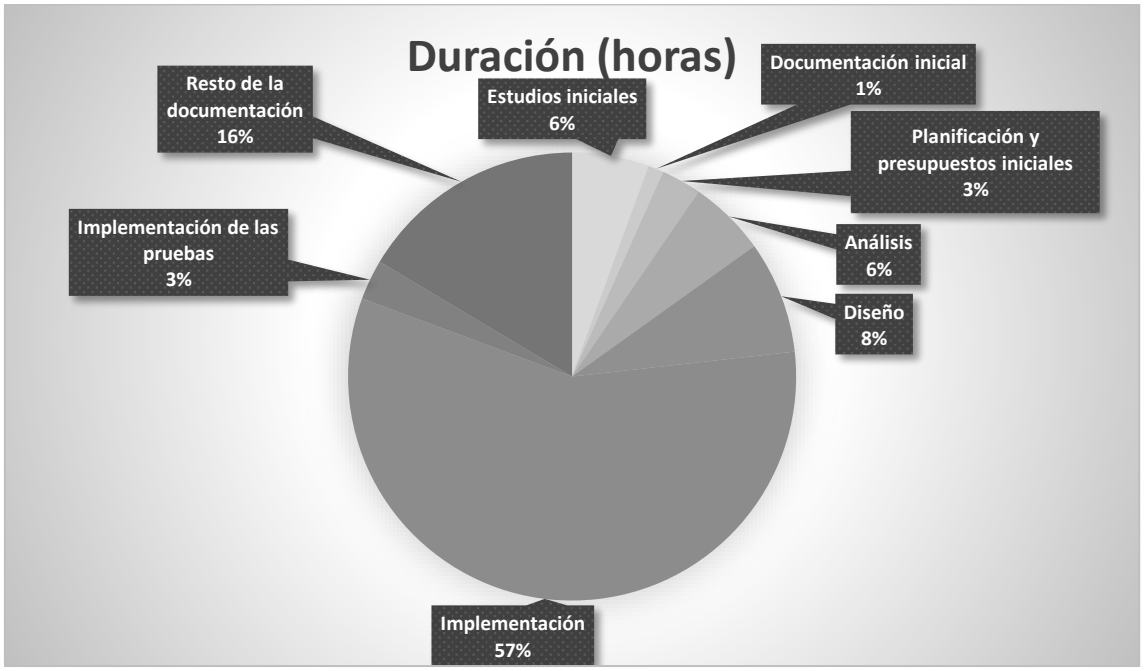



*Figura 11.1: Comparativa de duración de las etapas del proyecto en la planificación inicial*

Para la planificación final se ha tenido en cuenta una jornada laboral de **20 horas a la semana**, es decir, 4 horas más semanales que las que se habían establecido en la planificación inicial.

Cabe mencionar que la duración de la realización del presupuesto final ha sido estimada, dado que para realizar dicho presupuesto es necesario tener antes la planificación inicial.

A continuación, se muestra una imagen con las actividades de más alto nivel (etapas) y los hitos, así como un diagrama de Gantt:

| | Núm de esquı | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras |
|---|---|---|---|---|---|---|
| 1 | 1 | **Realización del TFG** | **450 horas** | **vie 01/02/19** | **dom 07/07/19** | |
| 2 | 1.1 | Inicio | 0 horas | vie 01/02/19 | vie 01/02/19 | |
| 3 | 1.2 | **Estudios iniciales** | **25 horas** | **vie 01/02/19** | **sáb 09/02/19** | **2** |
| 11 | 1.3 | Terminados los estudios iniciales | 0 horas | sáb 09/02/19 | sáb 09/02/19 | 3 |
| 12 | 1.4 | **Documentación inicial** | **5 horas** | **sáb 09/02/19** | **dom 10/02/19** | **11** |
| 17 | 1.5 | Terminada la documentación inicial | 0 horas | dom 10/02/19 | dom 10/02/19 | 12 |
| 18 | 1.6 | **Planificación y presupuestos iniciales** | **13 horas** | **dom 10/02/19** | **sáb 16/02/19** | **17** |
| 24 | 1.7 | Terminada la planificación y los presupuestos iniciales | 0 horas | sáb 16/02/19 | sáb 16/02/19 | 18 |
| 25 | 1.8 | **Análisis** | **25 horas** | **sáb 16/02/19** | **dom 24/02/19** | **24** |
| 37 | 1.9 | Terminado el análisis | 0 horas | dom 24/02/19 | dom 24/02/19 | 25 |
| 38 | 1.10 | **Diseño** | **36,5 horas** | **dom 24/02/19** | **sáb 09/03/19** | **37** |
| 54 | 1.11 | Terminado el diseño | 0 horas | sáb 09/03/19 | sáb 09/03/19 | 38 |
| 55 | 1.12 | **Implementación** | **258,5 horas** | **sáb 09/03/19** | **sáb 08/06/19** | **54** |
| 89 | 1.13 | Terminada la implementación | 0 horas | sáb 08/06/19 | sáb 08/06/19 | 55 |
| 90 | 1.14 | **Implementación de las pruebas** | **13 horas** | **sáb 08/06/19** | **mar 11/06/19** | **89** |
| 97 | 1.15 | Terminada la implementación de las pruebas | 0 horas | mar 11/06/19 | mar 11/06/19 | 90 |
| 98 | 1.16 | **Resto de la documentación** | **74 horas** | **mié 12/06/19** | **dom 07/07/19** | **97** |
| 130 | 1.17 | Terminada el resto de la documentación | 0 horas | dom 07/07/19 | dom 07/07/19 | 98 |
| 131 | 2 | Terminado el TFG | 0 horas | dom 07/07/19 | dom 07/07/19 | 1 |

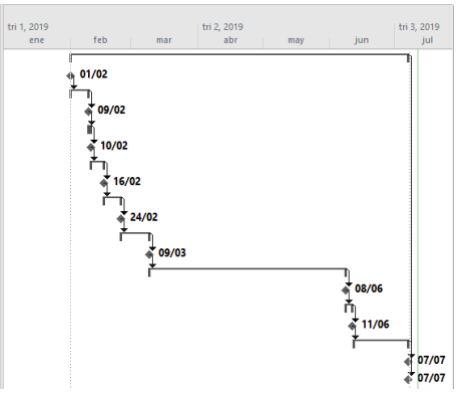



*Figura 11.2: Actividades de alto nivel y diagrama de Gantt de la planificación final*

Las filas en azul son las actividades de más alto nivel, y las filas en verde son los hitos.

A continuación, se muestra el WBS detallado[3], con las actividades de cada una de las etapas:

### 11.1.1 Estudios iniciales

| | Núm de esquı | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 3 | 1.2 | **Estudios iniciales** | **25 horas** | **vie 01/02/19** | **sáb 09/02/19** | **2** | | **25 horas** | **No** |
| 4 | 1.2.1 | Definición de los objetivos del proyecto | 3 horas | vie 01/02/19 | sáb 02/02/19 | | Desarrollador fullstack | 3 horas | No |
| 5 | 1.2.2 | Formación en Sistemas de Recuperación de información y NLP | 6 horas | sáb 02/02/19 | dom 03/02/19 | 4 | Desarrollador fullstack | 6 horas | No |
| 6 | 1.2.3 | Lectura de papers sobre modelos para la extracción de topics y clasificación de texto | 8 horas | dom 03/02/19 | mié 06/02/19 | 5 | Desarrollador fullstack | 8 horas | No |
| 7 | 1.2.4 | Lectura de papers sobre modelos para la generación de resumenes de textos | 2 horas | mié 06/02/19 | jue 07/02/19 | 6 | Desarrollador fullstack | 2 horas | No |
| 8 | 1.2.5 | Búsqueda y selección del dataset | 3 horas | jue 07/02/19 | vie 08/02/19 | 7 | Desarrollador fullstack | 3 horas | No |
| 9 | 1.2.6 | Investigación de librerías para la generación de los modelos de aprendizaje automático | 1 hora | sáb 09/02/19 | sáb 09/02/19 | 8 | Desarrollador fullstack | 1 hora | No |
| 10 | 1.2.7 | Comparación y selección de los lenguajes, herramientas, librerías y frameworks a utilizar para implementar el software del proyecto | 2 horas | sáb 09/02/19 | sáb 09/02/19 | 9 | Desarrollador fullstack | 2 horas | No |
| 11 | 1.3 | Terminados los estudios iniciales | 0 horas | sáb 09/02/19 | sáb 09/02/19 | 3 | | 0 horas | Sí |

*Figura 11.3: Actividades de la etapa "Estudios iniciales" en la planificación final*

[3] El WBS completo se encuentra en el archivo adjunto *Planificación final.mpp.*

### 11.1.2 Documentación inicial

| | Núm de esqui | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 12 | **1.4** | **Documentación inicial** | **5 horas** | **sáb 09/02/19** | **dom 10/02/19** | **11** | | **5 horas** | **No** |
| 13 | **1.4.1** | **Introducción** | **5 horas** | **sáb 09/02/19** | **dom 10/02/19** | | | **5 horas** | **No** |
| 14 | 1.4.1.1 | Justificación del proyecto | 2 horas | sáb 09/02/19 | sáb 09/02/19 | | Desarrollador fullstack | 2 horas | No |
| 15 | 1.4.1.2 | Objetivos del proyecto | 0,5 horas | dom 10/02/19 | dom 10/02/19 | 14 | Desarrollador fullstack | 0,5 horas | No |
| 16 | 1.4.1.3 | Estudio de la situación actual | 2,5 horas | dom 10/02/19 | dom 10/02/19 | 15 | Desarrollador fullstack | 2,5 horas | No |
| 17 | 1.5 | Terminada la documentación inicial | 0 horas | dom 10/02/19 | dom 10/02/19 | 12 | | 0 horas | Sí |

*Figura 11.4: Actividades de la etapa "Documentación inicial" en la planificación final*

### 11.1.3 Planificación y presupuestos iniciales

| | Núm de esqui | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 18 | **1.6** | **Planificación y presupuestos iniciales** | **13 horas** | **dom 10/02/19** | **sáb 16/02/19** | **17** | | **13 horas** | **No** |
| 19 | **1.6.1** | **Planificación inicial** | **5 horas** | **dom 10/02/19** | **mar 12/02/19** | | | **5 horas** | **No** |
| 20 | 1.6.1.1 | Creación de la Estructura de Desglose de Trabajo (WBS) | 4 horas | dom 10/02/19 | lun 11/02/19 | | Desarrollador fullstack | 4 horas | No |
| 21 | 1.6.1.2 | Estimación de actividades | 1 hora | mar 12/02/19 | mar 12/02/19 | 20 | Desarrollador fullstack | 1 hora | No |
| 22 | **1.6.2** | **Presupuesto inicial** | **8 horas** | **mar 12/02/19** | **sáb 16/02/19** | **19** | | **8 horas** | **No** |
| 23 | 1.6.2.1 | Presupuesto de costes inicial | 8 horas | mar 12/02/19 | sáb 16/02/19 | | Desarrollador fullstack | 8 horas | No |
| 24 | 1.7 | Terminada la planificación y los presupuestos iniciales | 0 horas | sáb 16/02/19 | sáb 16/02/19 | 18 | | 0 horas | Sí |

*Figura 11.5: Actividades de la etapa "Planificación y presupuestos iniciales" en la planificación final*

### 11.1.4 Análisis

| | Núm de esqui | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 25 | **1.8** | **Análisis** | **25 horas** | **sáb 16/02/19** | **dom 24/02/19** | **24** | | **25 horas** | **No** |
| 26 | 1.8.1 | Definición del alcance del sistema | 0,5 horas | sáb 16/02/19 | sáb 16/02/19 | | Desarrollador fullstack | 0,5 horas | No |
| 27 | **1.8.2** | **Requisitos** | **8,5 horas** | **sáb 16/02/19** | **dom 17/02/19** | **26** | | **8,5 horas** | **No** |
| 28 | 1.8.2.1 | Obtención de requisitos | 3,5 horas | sáb 16/02/19 | sáb 16/02/19 | | Desarrollador fullstack | 3,5 horas | No |
| 29 | 1.8.2.2 | Especificación de requisitos | 5 horas | dom 17/02/19 | dom 17/02/19 | 28 | Desarrollador fullstack | 5 horas | No |
| 30 | 1.8.3 | Casos de Uso | 3 horas | lun 18/02/19 | mar 19/02/19 | 27 | Desarrollador fullstack | 3 horas | No |
| 31 | 1.8.4 | Identificación de subsistemas | 0,5 horas | mar 19/02/19 | mar 19/02/19 | 30 | Desarrollador fullstack | 0,5 horas | No |
| 32 | 1.8.5 | Diagrama de clases preliminar del análisis | 6 horas | mar 19/02/19 | vie 22/02/19 | 31 | Desarrollador fullstack | 6 horas | No |
| 33 | **1.8.6** | **Análisis de interfaces de usuario** | **5,5 horas** | **vie 22/02/19** | **sáb 23/02/19** | **32** | | **5,5 horas** | **No** |
| 34 | 1.8.6.1 | Prototipos de pantallas a papel | 4 horas | vie 22/02/19 | sáb 23/02/19 | | Desarrollador fullstack | 4 horas | No |
| 35 | 1.8.6.2 | Pasar los prototipos de pantallas en papel a una herramienta | 1,5 horas | sáb 23/02/19 | sáb 23/02/19 | 34 | Desarrollador fullstack | 1,5 horas | No |
| 36 | 1.8.7 | Especificación del plan de pruebas | 1 hora | dom 24/02/19 | dom 24/02/19 | 33 | Desarrollador fullstack | 1 hora | No |
| 37 | 1.9 | Terminado el análisis | 0 horas | dom 24/02/19 | dom 24/02/19 | 25 | | 0 horas | Sí |

*Figura 11.6: Actividades de la etapa "Análisis" en la planificación final*

### 11.1.5 Diseño

| | Núm de esquí | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 38 | **1.10** | **Diseño** | **36,5 horas** | **dom 24/02/19** | **sáb 09/03/19** | **37** | | **36,5 horas** | **No** |
| 39 | **1.10.1** | **Arquitectura del sistema** | **6 horas** | **dom 24/02/19** | **lun 25/02/19** | | | **6 horas** | **No** |
| 40 | **1.10.1.1** | **Diagrama de componentes** | **4 horas** | **dom 24/02/19** | **dom 24/02/19** | | | **4 horas** | **No** |
| 41 | 1.10.1.1 | Subsistema Topics and Summary | 2,5 horas | dom 24/02/19 | dom 24/02/19 | | Desarrollador fullstack | 2,5 horas | No |
| 42 | 1.10.1.1 | Subsistema Web backend | 1 hora | dom 24/02/19 | dom 24/02/19 | 41 | Desarrollador fullstack | 1 hora | No |
| 43 | 1.10.1.1 | Subsistema Web frontend | 0,5 horas | dom 24/02/19 | dom 24/02/19 | 42 | Desarrollador fullstack | 0,5 horas | No |
| 44 | 1.10.1.2 | Diagramas de despliegue | 2 horas | lun 25/02/19 | lun 25/02/19 | 40 | Desarrollador fullstack | 2 horas | No |
| 45 | **1.10.2** | **Diseño de clases** | **19,5 horas** | **mar 26/02/19** | **lun 04/03/19** | **39** | | **19,5 horas** | **No** |
| 46 | 1.10.2.1 | Identificación de las clases, sus detalles y sus relaciones | 10 horas | mar 26/02/19 | sáb 02/03/19 | | Desarrollador fullstack | 10 horas | No |
| 47 | **1.10.2.2** | **Diagramas de clases** | **9,5 horas** | **sáb 02/03/19** | **lun 04/03/19** | **46** | | **9,5 horas** | **No** |
| 48 | 1.10.2.2 | Subsistema Topics and Summary | 3,5 horas | sáb 02/03/19 | dom 03/03/19 | | Desarrollador fullstack | 3,5 horas | No |
| 49 | 1.10.2.2 | Subsistema Web backend | 2 horas | dom 03/03/19 | dom 03/03/19 | 48 | Desarrollador fullstack | 2 horas | No |
| 50 | 1.10.2.2 | Subsistema Web frontend | 4 horas | dom 03/03/19 | lun 04/03/19 | 49 | Desarrollador fullstack | 4 horas | No |
| 51 | **1.10.3** | **Diseño de la interfaz** | **6 horas** | **lun 04/03/19** | **jue 07/03/19** | **45** | | **6 horas** | **No** |
| 52 | 1.10.3.1 | Prototipo interactivo de las pantallas con FluidIU | 6 horas | lun 04/03/19 | jue 07/03/19 | | Desarrollador fullstack | 6 horas | No |
| 53 | 1.10.4 | Especificación técnica del plan de pruebas | 5 horas | jue 07/03/19 | sáb 09/03/19 | 51 | Desarrollador fullstack | 5 horas | No |
| 54 | 1.11 | Terminado el diseño | 0 horas | sáb 09/03/19 | sáb 09/03/19 | 38 | | 0 horas | Sí |

*Figura 11.7: Actividades de la etapa "Diseño" en la planificación final*

### 11.1.6 Implementación

| | Núm de esquí | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 55 | **1.12** | **Implementación** | **258,5 horas** | **sáb 09/03/19** | **sáb 08/06/19** | **54** | | **258,5 horas** | **No** |
| 56 | 1.12.1 | Formación en tecnologías Python y librerías Data science/NLP | 4 horas | sáb 09/03/19 | dom 10/03/19 | | Desarrollador fullstack | 4 horas | No |
| 57 | **1.12.2** | **Implementación del subsistema "Topics and Summary"** | **122 horas** | **dom 10/03/19** | **dom 21/04/19** | **56** | | **122 horas** | **No** |
| 58 | 1.12.2.1 | Componente datasets | 15 horas | dom 10/03/19 | sáb 16/03/19 | | Desarrollador fullstack | 15 horas | No |
| 59 | 1.12.2.2 | Componente preprocessing | 35 horas | sáb 16/03/19 | jue 28/03/19 | 58 | Desarrollador fullstack | 35 horas | No |
| 60 | 1.12.2.3 | Componente embeddings | 3 horas | jue 28/03/19 | vie 29/03/19 | 59 | Desarrollador fullstack | 3 horas | No |
| 61 | **1.12.2.4** | **Componente models** | **44 horas** | **vie 29/03/19** | **sáb 13/04/19** | **60** | | **44 horas** | **No** |
| 62 | **1.12.2.4** | **Componente topics** | **40 horas** | **vie 29/03/19** | **vie 12/04/19** | | | **40 horas** | **No** |
| 63 | 1.12.2.4 | Implementación de modelos de aprendizaje automático para la extracción de topics | 20 horas | vie 29/03/19 | vie 05/04/19 | | Desarrollador fullstack | 20 horas | No |
| 64 | 1.12.2.4 | Implementación de las interfaces del componente topics | 20 horas | vie 05/04/19 | vie 12/04/19 | 63 | Desarrollador fullstack | 20 horas | No |
| 65 | 1.12.2.4 | Componente summarization | 4 horas | vie 12/04/19 | sáb 13/04/19 | 62 | Desarrollador fullstack | 4 horas | No |
| 66 | 1.12.2.5 | Componente visualizations | 10 horas | sáb 13/04/19 | mar 16/04/19 | 61 | Desarrollador fullstack | 10 horas | No |
| 67 | 1.12.2.6 | Demo con la funcionalidad del subsistema | 5 horas | mar 16/04/19 | vie 19/04/19 | 66 | Desarrollador fullstack | 5 horas | No |
| 68 | 1.12.2.7 | Transformación del subsistema en una librería Python instalable | 4 horas | vie 19/04/19 | sáb 20/04/19 | 67 | Desarrollador fullstack | 4 horas | No |
| 69 | 1.12.2.8 | Implementación de los ficheros Dockerfile y Vagrantfile | 6 horas | sáb 20/04/19 | dom 21/04/19 | 68 | Desarrollador fullstack | 6 horas | No |
| 70 | 1.12.3 | Pruebas unitarias del subsistema "Topics and Summary" | 8 horas | dom 21/04/19 | jue 25/04/19 | 57 | Desarrollador fullstack | 8 horas | No |
| 71 | 1.12.4 | Generación y comparativa de varios modelos de extracción de topics, y selección del modelo final | 14 horas | jue 25/04/19 | lun 29/04/19 | 70 | Desarrollador fullstack | 14 horas | No |
| 72 | 1.12.5 | Formación en Flask | 2 horas | lun 29/04/19 | mar 30/04/19 | 71 | Desarrollador fullstack | 2 horas | No |
| 73 | **1.12.6** | **Implementación del subsistema "Web backend"** | **26 horas** | **mar 30/04/19** | **vie 10/05/19** | **72** | | **26 horas** | **No** |
| 74 | 1.12.6.1 | Componente params | 2,5 horas | mar 30/04/19 | mié 01/05/19 | | Desarrollador fullstack | 2,5 horas | No |
| 75 | 1.12.6.2 | Componente wrapper | 8,5 horas | mié 01/05/19 | sáb 04/05/19 | 74 | Desarrollador fullstack | 8,5 horas | No |
| 76 | 1.12.6.3 | Componente apis.user | 7 horas | sáb 04/05/19 | lun 06/05/19 | 75 | Desarrollador fullstack | 7 horas | No |
| 77 | 1.12.6.4 | Implementación del fichero Dockerfile | 6 horas | lun 06/05/19 | jue 09/05/19 | 76 | Desarrollador fullstack | 6 horas | No |
| 78 | 1.12.6.5 | Despliegue del subsistema | 2 horas | jue 09/05/19 | vie 10/05/19 | 77 | Desarrollador fullstack | 2 horas | No |
| 79 | 1.12.7 | Pruebas unitarias del subsistema "Web backend" | 2 horas | vie 10/05/19 | sáb 11/05/19 | 73 | Desarrollador fullstack | 2 horas | No |
| 80 | 1.12.8 | Formación en Angular | 8 horas | sáb 11/05/19 | dom 12/05/19 | 79 | Desarrollador fullstack | 8 horas | No |
| 81 | **1.12.9** | **Implementación del subsistema "Web frontend"** | **72,5 horas** | **dom 12/05/19** | **sáb 08/06/19** | **80** | | **72,5 horas** | **No** |
| 82 | 1.12.9.1 | Componente topics | 30 horas | dom 12/05/19 | vie 24/05/19 | | Desarrollador fullstack | 30 horas | No |
| 83 | 1.12.9.2 | Componente text | 35 horas | vie 24/05/19 | mar 04/06/19 | 82 | Desarrollador fullstack | 35 horas | No |
| 84 | 1.12.9.3 | Componente app | 1 hora | mar 04/06/19 | mié 05/06/19 | 83 | Desarrollador fullstack | 1 hora | No |
| 85 | 1.12.9.4 | Componente header | 1 hora | mié 05/06/19 | mié 05/06/19 | 84 | Desarrollador fullstack | 1 hora | No |
| 86 | 1.12.9.5 | Componente footer | 0,5 horas | mié 05/06/19 | mié 05/06/19 | 85 | Desarrollador fullstack | 0,5 horas | No |
| 87 | 1.12.9.6 | Implementación del fichero Dockerfile | 4 horas | jue 06/06/19 | vie 07/06/19 | 86 | Desarrollador fullstack | 4 horas | No |
| 88 | 1.12.9.7 | Despliegue del subsistema | 1 hora | sáb 08/06/19 | sáb 08/06/19 | 87 | Desarrollador fullstack | 1 hora | No |
| 89 | 1.13 | Terminada la implementación | 0 horas | sáb 08/06/19 | sáb 08/06/19 | 55 | | 0 horas | Sí |

*Figura 11.8: Actividades de la etapa "Implementación" en la planificación final*

### 11.1.7 Implementación de las pruebas

| | Núm de esque | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 90 | **1.14** | **Implementación de las pruebas** | **13 horas** | **sáb 08/06/19** | **mar 11/06/19** | **89** | | **13 horas** | **No** |
| 91 | **1.14.1** | **Pruebas de usabilidad** | **5 horas** | **sáb 08/06/19** | **dom 09/06/19** | | | **5 horas** | **No** |
| 92 | 1.14.1.1 | Diseñar los cuestionarios | 1 hora | sáb 08/06/19 | sáb 08/06/19 | | Desarrollador fullstack | 1 hora | No |
| 93 | 1.14.1.2 | Realizar las pruebas de usabilidad a los 3 usuarios | 3 horas | sáb 08/06/19 | sáb 08/06/19 | 92 | Desarrollador fullstack | 3 horas | No |
| 94 | 1.14.1.3 | Analizar los resultados y sacar conclusiones | 1 hora | dom 09/06/19 | dom 09/06/19 | 93 | Desarrollador fullstack | 1 hora | No |
| 95 | 1.14.2 | Pruebas de adaptabilidad | 2 horas | dom 09/06/19 | dom 09/06/19 | 91 | Desarrollador fullstack | 2 horas | No |
| 96 | 1.14.3 | Análisis estático de código | 6 horas | dom 09/06/19 | mar 11/06/19 | 95 | Desarrollador fullstack | 6 horas | No |
| 97 | 1.15 | Terminada la implementación de las pruebas | 0 horas | mar 11/06/19 | mar 11/06/19 | 90 | | 0 horas | Sí |

*Figura 11.9: Actividades de la etapa "Implementación de las pruebas" en la planificación final*

La implementación de las pruebas unitarias se ha añadido finalmente dentro de la etapa de implementación, y no en esta etapa, en la que se incluyen el resto de pruebas.

### 11.1.8 Resto de la documentación

| | Núm de esque | Nombre de tarea | Duración | Comienzo | Fin | Predecesoras | Nombres de los recursos | Trabajo | Hito |
|---|---|---|---|---|---|---|---|---|---|
| 98 | **1.16** | **Resto de la documentación** | **74 horas** | **mié 12/06/19** | **dom 07/07/19** | **97** | | **74 horas** | **No** |
| 99 | **1.16.1** | **Documentación de la implementación del sistema** | **4 horas** | **mié 12/06/19** | **jue 13/06/19** | | | **4 horas** | **No** |
| 100 | 1.16.1.1 | Estándares y Normas seguidos | 0,25 horas | mié 12/06/19 | mié 12/06/19 | | Desarrollador fullstack | 0,25 horas | No |
| 101 | 1.16.1.2 | Lenguajes de programación | 0,25 horas | mié 12/06/19 | mié 12/06/19 | 100 | Desarrollador fullstack | 0,25 horas | No |
| 102 | 1.16.1.3 | Herramientas y Programas usados para el desarrollo | 0,5 horas | mié 12/06/19 | mié 12/06/19 | 101 | Desarrollador fullstack | 0,5 horas | No |
| 103 | 1.16.1.4 | Creación del sistema | 3 horas | mié 12/06/19 | jue 13/06/19 | 102 | Desarrollador fullstack | 3 horas | No |
| 104 | **1.16.2** | **Documentación del desarrollo de las pruebas** | **2 horas** | **vie 14/06/19** | **vie 14/06/19** | **99** | | **2 horas** | **No** |
| 105 | 1.16.2.1 | Pruebas unitarias automatizadas | 2 horas | vie 14/06/19 | vie 14/06/19 | | Desarrollador fullstack | 2 horas | No |
| 106 | **1.16.3** | **Manuales del sistema** | **41 horas** | **sáb 15/06/19** | **sáb 29/06/19** | **104** | | **41 horas** | **No** |
| 107 | 1.16.3.1 | Documentación del subsistema "Topics and Summary" | 20 horas | sáb 15/06/19 | vie 21/06/19 | | Desarrollador fullstack | 20 horas | No |
| 108 | 1.16.3.2 | Documentación del subsistema "Web backend" | 10 horas | sáb 22/06/19 | dom 23/06/19 | 107 | Desarrollador fullstack | 10 horas | No |
| 109 | 1.16.3.3 | Documentación de la API REST con Swagger | 3 horas | lun 24/06/19 | mar 25/06/19 | 108 | Desarrollador fullstack | 3 horas | No |
| 110 | 1.16.3.4 | Documentación del subsistema "Web frontend" | 8 horas | mar 25/06/19 | sáb 29/06/19 | 109 | Desarrollador fullstack | 8 horas | No |
| 111 | 1.16.4 | Aspectos teóricos | 4 horas | sáb 29/06/19 | sáb 29/06/19 | 106 | Desarrollador fullstack | 4 horas | No |
| 112 | **1.16.5** | **Conclusiones y ampliaciones** | **3,5 horas** | **dom 30/06/19** | **dom 30/06/19** | **111** | | **3,5 horas** | **No** |
| 113 | 1.16.5.1 | Conclusiones | 1,5 horas | dom 30/06/19 | dom 30/06/19 | | Desarrollador fullstack | 1,5 horas | No |
| 114 | 1.16.5.2 | Ampliaciones | 2 horas | dom 30/06/19 | dom 30/06/19 | 113 | Desarrollador fullstack | 2 horas | No |
| 115 | 1.16.6 | Memoria del proyecto | 3 horas | dom 30/06/19 | lun 01/07/19 | 112 | Desarrollador fullstack | 3 horas | No |
| 116 | 1.16.7 | Referencias bibliográficas | 2 horas | lun 01/07/19 | mar 02/07/19 | 115 | Desarrollador fullstack | 2 horas | No |
| 117 | **1.16.8** | **Apéndices** | **7,5 horas** | **mar 02/07/19** | **sáb 06/07/19** | **116** | | **7,5 horas** | **No** |
| 118 | 1.16.8.1 | Glosario | 2 horas | mar 02/07/19 | mié 03/07/19 | | Desarrollador fullstack | 2 horas | No |
| 119 | 1.16.8.2 | Contenido entregado en el archivo adjunto | 0,25 horas | mié 03/07/19 | mié 03/07/19 | 118 | Desarrollador fullstack | 0,25 horas | No |
| 120 | 1.16.8.3 | Código fuente | 0,25 horas | mié 03/07/19 | mié 03/07/19 | 119 | Desarrollador fullstack | 0,25 horas | No |
| 121 | 1.16.8.4 | Documentación del presupuesto de costes inicial (detallado) | 4 horas | jue 04/07/19 | vie 05/07/19 | 120 | Desarrollador fullstack | 4 horas | No |
| 122 | 1.16.8.5 | Documentación del presupuesto de costes final (detallado) | 1 hora | sáb 06/07/19 | sáb 06/07/19 | 121 | Desarrollador fullstack | 1 hora | No |
| 123 | 1.16.9 | Agradecimientos, resumen y keywords | 2 horas | sáb 06/07/19 | sáb 06/07/19 | 117 | Desarrollador fullstack | 2 horas | No |
| 124 | **1.16.10** | **Planificación y presupuestos finales** | **5 horas** | **sáb 06/07/19** | **dom 07/07/19** | **123** | | **5 horas** | **No** |
| 125 | **1.16.10.** | **Planificación final** | **3 horas** | **sáb 06/07/19** | **dom 07/07/19** | | | **3 horas** | **No** |
| 126 | 1.16.10. | Modificación de la Estructura de Desglose de Trabajo (WBS) | 2 horas | sáb 06/07/19 | sáb 06/07/19 | | Desarrollador fullstack | 2 horas | No |
| 127 | 1.16.10. | Actualización de las estimaciones de las actividades con los tiempos reales | 1 hora | dom 07/07/19 | dom 07/07/19 | 126 | Desarrollador fullstack | 1 hora | No |
| 128 | **1.16.10.** | **Presupuesto final** | **2 horas** | **dom 07/07/19** | **dom 07/07/19** | **125** | | **2 horas** | **No** |
| 129 | 1.16.10. | Presupuesto de costes final | 2 horas | dom 07/07/19 | dom 07/07/19 | | Desarrollador fullstack | 2 horas | No |
| 130 | 1.17 | Terminada el resto de la documentación | 0 horas | dom 07/07/19 | dom 07/07/19 | 98 | | 0 horas | Sí |
| 131 | 2 | Terminado el TFG | 0 horas | dom 07/07/19 | dom 07/07/19 | 1 | | 0 horas | Sí |

*Figura 11.10: Actividades de la etapa "Resto de la documentación" en la planificación final*

## 11.2 Presupuesto de Costes Final (resumido)

A continuación, se muestra el presupuesto de costes resumido, basado en la planificación anterior:

| Presupuesto de costes | | |
|---|---|---|
| ***Cód.*** | ***Partida*** | ***Total*** |
| 01 | Estudios iniciales y planificación inicial | 1.425,00 € |
| 02 | Análisis y diseño del sistema | 2.306,25 € |
| 03 | Implementación y despliegue del sistema | 10.312,45 € |
| 04 | Documentación | 2.962,50 € |
| 05 | Otros costes | 1.354,20 € |
| | **Total Coste** | **18.360,40 €** |

*Tabla 2: Presupuesto de costes final (resumido)*

**Nota:** El coste que figura aquí no incluye el IVA.

El coste que nos salía en el presupuesto inicial era de **12.483,00€**. Por tanto, el proyecto ha tenido un coste total **5.877,40€** superior al planificado inicialmente.

En el apéndice *Presupuesto de Costes Final (detallado)* se detalla cómo se ha calculado este presupuesto de costes final.

# Capítulo 12. Referencias Bibliográficas


[1] R. Francois Chaubard y R. S. Mundra, «**CS 224D: Deep Learning for NLP - Lecture Notes: Part I**», 2016. http://web.stanford.edu/class/cs224d/lecture_notes/notes1.pdf. [Consultado: 07-jul-2019].

[2] «**Natural language processing**», 2019. https://en.wikipedia.org/wiki/Natural_language_processing. [Consultado: 07-jul-2019].

[3] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, y J. Dean, «**Distributed Representations of Words and Phrases and their Compositionality**», 2013.

[4] «**Word embedding**», 2019. https://es.wikipedia.org/wiki/Word_embedding. [Consultado: 07-jul-2019].

[5] M. Naili, A. H. Chaibi, y H. H. Ben Ghezala, «**Comparative study of word embedding methods in topic segmentation**», *Procedia Computer Science*, 2017.

[6] T. Mikolov, K. Chen, G. Corrado, y J. Dean, «**Efficient Estimation of Word Representations in Vector Space**», 2013.

[7] J. Pennington, R. Socher, y C. D. Manning, «**GloVe: Global Vectors for Word Representation**», 2014.

[8] T. Ganegedara, «**Intuitive Guide to Latent Dirichlet Allocation**», 2018. https://towardsdatascience.com/light-on-math-machine-learning-intuitive-guide-to-latent-dirichlet-allocation-437c81220158. [Consultado: 07-jul-2019].

[9] D. M. Blei, A. Y. Ng, y M. I. Jordan, «**Latent Dirichlet Allocation**», *Journal of Machine Learning Research*, vol. 3, pp. 993-1022, 2003.

[10] R. Řehůřek y P. Sojka, «**Software Framework for Topic Modelling with Large Corpora**», en *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, 2010, pp. 45-50.

[11] A. K. McCallum, «**MALLET: A Machine Learning for Language Toolkit**», 2002. http://mallet.cs.umass.edu.

[12] «**PageRank**», 2019. https://es.wikipedia.org/wiki/PageRank. [Consultado: 07-jul-2019].

[13] P. Joshi, «**An Introduction to Text Summarization using the TextRank Algorithm**», 2018. https://www.analyticsvidhya.com/blog/2018/11/introduction-text-summarization-textrank-python/. [Consultado: 07-jul-2019].

[14] R. Mihalcea y P. Tarau, «**TextRank: Bringing Order into Texts**», en *Conference on Empirical Methods in Natural Language Processing (EMNLP), 2004, Barcelona, Spain*, 2004, pp. 404-411.

[15] «**Transferencia de Estado Representacional**», 2018. https://es.wikipedia.org/wiki/Transferencia_de_Estado_Representacional. [Consultado: 07-jul-2019].

[16] «**Single-page application**», 2019. https://en.wikipedia.org/wiki/Single-page_application. [Consultado: 07-jul-2019].

[17] «**FluidUI**». https://www.fluidui.com. [Consultado: 07-jul-2019].

[18] G. van Rossum, B. Warsaw, y N. Coghlan, «**PEP 8 -- Style Guide for Python Code**», 2001. https://www.python.org/dev/peps/pep-0008/. [Consultado: 07-jul-2019].

[19] «**Angular Style Guide**». https://angular.io/guide/styleguide. [Consultado: 07-jul-2019].

[20] «**ROUGE (metric)**», 2019. https://en.wikipedia.org/wiki/ROUGE_(metric). [Consultado: 07-jul-2019].

[21] M. Allahyari *et al.*, «**Text Summarization Techniques: A Brief Survey**», 2017.

[22] «**Angular Internationalization (i18n)**». https://angular.io/guide/i18n. [Consultado: 07-jul-2019].

[23] A. Kirkpatrick, J. O. Connor, A. Campbell, y M. Cooper, «**Web Content Accessibility Guidelines (WCAG) 2.1**», 2018. https://www.w3.org/TR/WCAG21/. [Consultado: 07-jul-2019].

[24] A. A. Juan Fuente y B. López Pérez, «**Guía de Aprendizaje de la asignatura de Dirección y Planificación de Proyectos Informáticos**», Universidad de Oviedo, 2019.

[25] «**Bases y tipos de cotización 2019**», 2019. http://www.seg-social.es/wps/portal/wss/internet/Trabajadores/CotizacionRecaudacionTrabajadores/36537. [Consultado: 07-jul-2019].

# Capítulo 13. Apéndices

## 13.1 Glosario y Diccionario de Datos

- **20 newsgroups:** Dataset compuesto por documentos de noticias en inglés, utilizado para entrenar el modelo de extracción de topics. Los documentos están divididos en 20 grupos.
- **Accesibilidad web:** Tiene como objetivo como objetivo lograr que las páginas web sean utilizables por el máximo número de personas, incluyendo personas con algún tipo de discapacidad.
- **Adaptabilidad web:** Tiene como objetivo adaptar la apariencia de las páginas web al dispositivo que se esté utilizando para visualizarlas.
- **Angular:** Framework para aplicaciones web, desarrollado en TypeScript, que permite desarrollar aplicaciones del tipo SPA.
- **API:** Application Programming Interface o Interfaz de programación de aplicaciones. Conjunto de funciones que ofrecen una cierta funcionalidad que puede ser utilizada por otro software.
- **AWS:** Amazon Web Services. Plataforma de computación en la nube.
- **Backend:** Parte de una aplicación software que procesa la información introducida por el usuario a través del frontend.
- **Batch:** Ejecución de un programa sin interacción con el usuario. Se utiliza en tareas repetitivas sobre grandes conjuntos de información.
- **CLI:** Command Line Interface o Interfaz de Línea de Comandos. Permite a los usuarios interaccionar con un software a través de una línea de texto simple.
- **CSS:** Cascading Style Sheets o Hojas de estilo en cascada. Lenguaje utilizado para definir la presentación visual de un documento escrito en lenguaje de marcado.
- **Dataframe:** Principal estructura de datos de la librería Pandas. Tiene formato tabular.
- **Dataset:** Conjunto de datos utilizado para entrenar modelos de aprendizaje automático.
- **Docker:** Plataforma para desarrolladores y administradores de sistemas, basada en contenedores.
- **Flask:** Microframework para desarrollo web en el lenguaje de programación Python.
- **Framework:** Marco de trabajo que sirve como base para la organización y desarrollo de software.
- **Frontend:** Parte de una aplicación software que interacciona con los usuarios. Se comunica con el backend.
- **Gensim:** Librería Python de aprendizaje automático enfocada en la agrupación de textos en clusters y en el análisis automático de temáticas presentes en documentos de texto.
- **GloVe:** Algoritmo de aprendizaje no supervisado que permite obtener representaciones vectoriales de palabras (es un método de word embedding). GloVe viene de *Global Vectors*, ya que el entrenamiento se realiza sobre estadísticas globales agregadas de coocurrencia de palabras.
- **Heroku:** Plataforma de servicios en la nube que permite desplegar, monitorizar y escalar aplicaciones fácilmente.

- **HTML:** Lenguaje de marcado estándar para el desarrollo de páginas web.
- **HTTP:** Protocolo de comunicación que permite las transferencias de información en la web.
- **Internacionalización:** Proceso de diseñar software de manera tal que pueda adaptarse a diferentes idiomas y regiones sin la necesidad de realizar cambios en el código fuente.
- **IDE:** Integrated Development Environment o Entorno de Desarrollo Integrado. Aplicación informática que facilita el desarrollo de software.
- **Java:** Lenguaje de programación de propósito general cuyo paradigma principal es el orientado a objetos.
- **JDK:** Java Development Kit. Software que provee herramientas de desarrollo para la creación de programas en Java.
- **Keyword:** Palabra clave perteneciente a un topic, y que describe en mayor o menor medida la temática de dicho topic.
- **LDA:** Latent Diriletch Allocation. Principal algoritmo utilizado en este proyecto para la extracción de topics de un conjunto de documentos de texto.
- **Mallet:** Librería Java de aprendizaje automático enfocada en el Procesamiento de Lenguaje Natural, la clasificación de texto, la identificación de topics y otros campos relacionados con el procesamiento de texto.
- **Microsoft Project:** Software de administración y gestión de proyectos, utilizado en este proyecto para realizar la planificación.
- **Modelo:** Representación de un conjunto de datos creada por un algoritmo de aprendizaje automático.
- **Ngram:** Conjunto de n elementos extraídos de un texto (sílabas, letras, palabras, …) que aparecen frecuentemente juntos, y que por separado no aparecen con tanta frecuencia.
- **NLP:** Natural Language Processing o Procesamiento de Lenguaje Natural. Rama de la inteligencia artificial, que engloba una serie de técnicas utilizadas para diseñar algoritmos que permitan a los ordenadores "entender" el lenguaje natural, con el fin de realizar alguna tarea.
- **Node.js:** Plataforma de software que permite ejecutar JavaScript del lado del servidor.
- **Nube de palabras:** Representación gráfica que muestra las keywords de un topic, en la que la probabilidad de las keywords define su tamaño (palabras con mayor probabilidad tienen mayor tamaño).
- **Path:** Ruta dentro de un sistema de archivos.
- **Preprocesado:** Etapa importante dentro del proceso de generación de modelos de aprendizaje automático, que consiste en "limpiar" el dataset para facilitar el trabajo a los algoritmos y conseguir que obtengan mejores resultados. El preprocesado que se puede aplicar es muy diverso, y dependerá en gran medida de la tarea que se esté realizando.
- **Python:** Lenguaje de programación interpretado y de propósito general, cuyos puntos fuertes son su sencillez y legibilidad. Es ampliamente utilizado en el área del aprendizaje automático.
- **REST:** Representational State Transfer. Se trata de un estilo de arquitectura software para sistemas hipermedia distribuidos, en el que los datos y la funcionalidad se consideran recursos, y se accede a ellos utilizando URIs.

- **Resumen extractivo:** Resumen consistente en un conjunto de frases extraídas directamente del texto original.
- **Scroll:** Movimiento en dos dimensiones de la ventana que se muestra en una interfaz de usuario.
- **SonarQube:** Herramienta de análisis estático de código.
- **SPA:** Single Page Application o Aplicación de página única. Se trata de un tipo de aplicación web en la que todos los contenidos de la misma se sirven en una única página.
- **TextRank:** Algoritmo utilizado para generar resúmenes de documentos de texto, basado en el algoritmo PageRank.
- **Topic:** Conjunto de keywords que representan una temática presente en el conjunto de documentos de texto.
- **Typescript:** Lenguaje de programación, que se caracteriza por ser un superconjunto tipado de JavaScript, y que se compila a JavaScript puro.
- **URI:** Uniform Resource Identifier o Identificador de Recursos Uniforme. Cadena de caracteres que identifica los recursos de una red de forma unívoca.
- **Vagrant:** Herramienta para la creación y manejo de máquinas virtuales.
- **WBS:** Work Breakdown Structure o Estructura de Desglose de Trabajo. Es la lista de actividades necesarias para completar un proyecto.
- **Word2Vec:** Grupo de modelos relacionados utilizados para obtener representaciones vectoriales de palabras (son métodos de word embedding). En concreto, los modelos que puede utilizar son: *Continuous Bag of Words* y *Continuous Skip-Gram*.
- **Word embedding:** Conjunto de modelos, técnicas y algoritmos de procesamiento de lenguaje natural cuyo objetivo es transformar las palabras (o frases en el caso de phrase embedding) en vectores de números reales de baja dimensión.

## 13.2 Contenido Entregado en el Archivo adjunto

A continuación, se realiza una breve descripción del contenido del Archivo adjunto.

| Directorio | Contenido |
|---|---|
| *./ Directorio raíz del Archivo adjunto* | Contiene un fichero README.md explicando toda esta estructura. |
| *./Topics and Summary* | Contiene las carpetas de los 3 subsistemas y los ficheros Docker necesarios para generar la imagen del subsistema *Web backend*. |
| *./Topics and Summary/topics_and_summary* | Contiene el código fuente y la documentación del subsistema *Topics and Summary*. Una explicación más en detalle de su contenido se encuentra en la ruta especificada en el apartado 9.3.1.1. |
| *./Topics and Summary/web_backend* | Contiene el código fuente y la documentación del subsistema *Web backend*. Una explicación más en detalle de su contenido se encuentra en la ruta especificada en el apartado 9.3.1.2. |
| *./Topics and Summary/web-frontend* | Contiene el código fuente y la documentación del subsistema *Web frontend*. Una explicación más en detalle de su contenido se encuentra en la ruta especificada en el apartado 9.3.1.3. |
| *./Diagramas* | Contiene los diagramas UML, tanto en formato PDF, como en el formato que genera la herramienta Umlet (.uxf). |
| *./Planificación y presupuesto* | Contiene la planificación y el presupuesto iniciales y finales, así como el modelo de empresa sobre el que se basan dichos presupuestos. |
| *./Planificación y presupuesto/Iniciales* | Contiene la planificación (en formato Microsoft Project) y el presupuesto iniciales. |
| *./Planificación y presupuesto/Finales* | Contiene la planificación (en formato Microsoft Project) y el presupuesto finales. |

## 13.3 Código Fuente

Como se menciona en el apartado anterior, la carpeta Topics and Summary contiene las carpetas de los 3 subsistemas. Dentro de cada una de ellas se encuentra el código fuente y la documentación de cada subsistema.

Una explicación detallada del contenido de los subsistemas Topics and Summary, Web backend y Web frontend se encuentra en la rutas especificadas en los apartados 9.3.1.1, 9.3.1.2 y 9.3.1.3 respectivamente.

# 13.4 Presupuesto de Costes Inicial (detallado)

El presupuesto se ha realizado basándose en lo explicado en [24]. Primeramente, se describe el modelo de empresa en el que se han basado los cálculos posteriores.

## 13.4.1 Modelo de empresa

El archivo adjunto *./Planificación y presupuesto/Modelo de empresa.xlsx* contiene toda la información y los cálculos relativos al modelo de empresa sobre el que se han basado parte de los cálculos del presupuesto de costes. En específico, este modelo de empresa define los perfiles necesarios para la realización del proyecto y sus precios/hora (sin beneficios). Se trata de un modelo de empresa simplificado, que describe una empresa "Orientada a proyectos".

A continuación, se explica dicho modelo de empresa.

### *13.4.1.1 Personal de la empresa*

Dadas las características del Trabajo Fin de Grado, se ha supuesto que la empresa cuenta con un único empleado: un desarrollador fullstack (el autor del TFG), con el siguiente sueldo y coste:

| Personal | Núm | Sueldo Bruto año | Coste Salarial Año | TOTAL |
|---|---|---|---|---|
| Desarrollador Fullstack | 1 | 21.000,00 € | 27.426,00 € | 27.426,00 € |
| **TOTAL** | **1** | | | **27.426,00 €** |

*Tabla 3: Personal, sus sueldos y sus costes*

El coste salarial anual se ha calculado incrementando un 30.6% al sueldo bruto anual. Dicho 30.6% incluye:

- El porcentaje de seguridad social que debe pagar la empresa por los trabajadores, que para el año 2019, en el caso de horas de trabajo comunes para empresas, es del 23.60% del sueldo bruto anual, tal como se indica en [25].
- El porcentaje de bonus que se añade al sueldo de los trabajadores, que en el caso de esta empresa suponemos que es del 5% del sueldo bruto anual.
- Los periodos de vacaciones. Suponemos que esto representa a la empresa un coste equivalente al 2% del sueldo bruto anual de los trabajadores.

Por tanto, según los cálculos, el coste total de los salarios de la empresa para un año es de 27.426,00 €.

### *13.4.1.2 Asignación de los costes salariales a costes directos e indirectos*

A continuación, se asignan los costes de los salarios a coste directos e indirectos, teniendo en cuenta el porcentaje de productividad de cada miembro del personal de la empresa.

La productividad indica el porcentaje de tiempo que un miembro del personal dedica a la realización de proyectos en la empresa. Se ha de tener en cuenta que la capacidad de producción de un recurso tiene algunos componentes que impiden una efectividad del 100%, como paros, interferencias, ...

| Personal | TOTAL | Productividad (%) | Coste directo | Coste indirecto (%) | Coste indirecto |
|---|---|---|---|---|---|
| Desarrollador Fullstack | 27.426,00 € | 85% | 23.312,10 € | 15% | 4.113,90 € |
| **TOTAL** | **27.426,00 €** | | **23.312,10 €** | | **4.113,90 €** |

*Tabla 4: Asignación de los costes salariales a costes directos e indirectos*

El desarrollador está siempre realizando proyectos, pero también tiene descansos, distracciones, auditorías de la empresa, reuniones con los jefes, actualizaciones del equipo de desarrollo, …, por lo que se ha supuesto una productividad del 85%.

#### 13.4.1.3 Otros costes indirectos

Además de los costes debidos a salarios, hay otros costes que tener en cuenta para calcular el tamaño de la empresa.

##### 13.4.1.3.1 Costes de servicios necesarios para el funcionamiento de la empresa

| Servicio | Coste mes | Coste año |
|---|---|---|
| Limpieza | 150,00 € | 1.800,00 € |
| Asesoría | 400,00 € | 4.800,00 € |
| Alquiler de las oficinas | 1.000,00 € | 12.000,00 € |
| Consumos de electricidad (excepto consumos para producción) | 250,00 € | 3.000,00 € |
| Consumos de combustible para calefacción | 100,00 € | 1.200,00 € |
| Consumos de agua | 75,00 € | 900,00 € |
| Honorarios de asesorías, auditorías y otros profesionales | 200,00 € | 2.400,00 € |
| Primas de seguros | 300,00 € | 3.600,00 € |
| Portes y gastos de transporte | 75,00 € | 900,00 € |
| Gastos en comunicaciones (Internet y telefonía) | 200,00 € | 2.400,00 € |
| Gastos en material de oficina | 50,00 € | 600,00 € |
| Gastos de correos y mensajería | 30,00 € | 360,00 € |
| Gastos financieros | 300,00 € | 3.600,00 € |
| **TOTAL** | **3.130,00 €** | **37.560,00 €** |

*Tabla 5: Lista de costes de servicios necesarios para el funcionamiento de la empresa*

#### 13.4.1.3.2 Costes de los medios de producción

Para poder realizar los proyectos de software, son necesarios diversos medios tecnológicos y herramientas de soporte. En este caso, se ha considerado un único equipo de desarrollo, que será utilizado por el desarrollador, así como ciertas licencias de software, que se exponen a continuación.

| Servicio | Unid | Precio | Coste Total | Coste Año | Tipo | Plazo |
|---|---|---|---|---|---|---|
| Portátil de desarrollo (Macbook Pro) | 1 | 2.249,00 € | 2.249,00 € | 1.124,50 € | Amortización | 2 |
| Licencia Microsoft Office 365 Empresa Premium | 1 | 152,46 € | 152,46 € | 152,46 € | Alquiler | - |
| Licencia Microsoft Project | 1 | 849,00 € | 849,00 € | 849,00 € | Alquiler | - |
| Licencia Virtual Box | 1 | 0,00 € | 0,00 € | 0,00 € | Alquiler | - |
| **TOTAL** | | | | **2.125,96 €** | | |

*Tabla 6: Lista de los costes de los medios de producción*

### 13.4.1.4 Coste total (costes directos + indirectos + beneficios)

Suponemos que los costes indirectos son todos aquellos costes que se tendrían en caso de que no se pudiera producir, además de aquellos recursos que se utilizan para cometidos distintos de los de producción.

Por tanto, los **costes indirectos** de la empresa se calculan sumando los costes indirectos de los salarios (4.113,90 €, obtenidos de la fila de totales, en la columna 'Coste indirecto', de la Tabla 4), y los otros costes indirectos (calculados en el apartado 13.4.1.3), que se desglosan en costes indirectos derivados de servicios para el funcionamiento de la empresa (37.560,00 €, obtenidos de la fila de totales, en la columna 'Coste año', de la Tabla 5) y en costes indirectos derivados de los medios de producción (2.125,96 €, obtenidos de la fila de totales de la Tabla 6). Esto suma un total de **43.799,86 €** en costes indirectos. La siguiente tabla muestra un resumen de lo mencionado anteriormente:

| Resumen costes indirectos | Importe |
|---|---|
| Costes indirectos de los salarios | 4.113,90 € |
| Costes indirectos derivados de servicios para el funcionamiento de la empresa | 37.560,00 € |
| Costes indirectos derivados de los medios de producción | 2.125,96 € |
| **TOTAL** | **43.799,86 €** |

*Tabla 7: Resumen de los costes indirectos*

Por otro lado, los **costes directos** son los costes directos derivados de los salarios (23.312,10 €, obtenidos de la fila de totales, en la columna 'Coste directo', de la Tabla 4).

Así pues, el coste total de la empresa en un año es la suma de costes directos e indirectos, esto es, **67.111,96 €**. A esta cantidad hay que sumar el beneficio esperado de la empresa.

Hemos supuesto que la empresa espera obtener un **beneficio del 25%** de la suma de costes directos e indirectos, esto es, **16.777,99 €**.

Sumados los costes más los beneficios, la empresa tendría una necesidad de facturación de **83.889,95 €**.

La siguiente tabla muestra un resumen de lo mencionado anteriormente:

| Concepto | Importe |
|---|---|
| Total de los costes directos | 23.312,10 € |
| Total de los costes indirectos | 43.799,86 € |
| Suma de los costes directos e indirectos | 67.111,96 € |
| Beneficio deseado (€) | 16.777,99 € |
| **Coste total** (suma de los costes directos, indirectos y beneficios) | 83.889,95 € |

*Tabla 8: Resumen de los costes del modelo de empresa*

### 13.4.1.5 Horas productivas y no productivas

De la misma manera que los costes salariales se han asignado a costes directos e indirectos, las horas de trabajo se pueden asignar a horas productivas y no productivas. Se han usado exactamente los mismos porcentajes de productividad que para los costes.

Se han asignado 2.008 horas de trabajo anuales por trabajador (según el calendario del 2019).

| Personal | Productividad (%) | Horas/año | Horas productivas /año (por persona) | Horas productivas (Total empresa) |
|---|---|---|---|---|
| Desarrollador Fullstack | 85% | 2008 | 1706,8 | 1706,80 |
| **TOTAL** | | | | **1706,80** |

*Tabla 9: Número de horas productivas por perfil y en total*

### 13.4.1.6 Precio hora

#### 13.4.1.6.1 Para el cliente

Con todos los datos anteriores, se puede establecer un precio por hora de trabajo a cada perfil, que permita cubrir todos los costes de la empresa, y que genere los beneficios deseados. Estos precios incluyen todos los costes indirectos.

Multiplicando el precio/hora de cada perfil por sus horas productivas, y sumando los precios que salen, obtenemos la cantidad que la empresa podrá facturar anualmente. Estos precios por hora solamente se utilizarán en los presupuestos por horas para el cliente.

Las siguientes tablas muestran los precios/hora asignados, y el margen entre el coste total y la facturación:

| Servicio | Precio/hora | Horas productivas (Total empresa) | Facturación |
|---|---|---|---|
| Desarrollador Fullstack | 50,00 € | 1.706,80 | 85.340,00 € |
| | | | **85.340,00 €** |

*Tabla 10: Precios por hora de trabajo y facturación total de la empresa*

| Concepto | Importe |
|---|---|
| Coste total (suma de los costes directos, indirectos y beneficios) | 83.889,95 € |
| Facturación posible en función de las horas de producción y de los precios por hora calculados | 85.340,00 € |
| Margen entre el coste total y la facturación | 1,70% |

*Tabla 11: Margen entre el coste total y la facturación*

#### 13.4.1.6.2 Para el cálculo de los costes del proyecto

Aquí se tienen en cuenta los precios/hora de cada perfil, sin incluir los beneficios. Es decir, a los precios/hora de la Tabla 10 se les quita el porcentaje de beneficio. En nuestro caso, esto consiste en multiplicar cada precio/hora de dicha tabla por (1-0,25), dado que la empresa espera obtener beneficios del 25%. Dichos costes se muestran en la siguiente tabla:

| Personal | Precio/hora (Sin beneficios) |
|---|---|
| Desarrollador Fullstack | 37,50 € |

*Tabla 12: Precio por hora a usar para el cálculo de los costes del proyecto*

## 13.4.2 Explicación del presupuesto de costes inicial

El presupuesto consta de las siguientes partidas:

1. **Partida 1:** Estudios iniciales y planificación inicial
2. **Partida 2:** Análisis y diseño del sistema
3. **Partida 3:** Implementación y despliegue del sistema
4. **Partida 4:** Documentación
5. **Partida 5:** Otros costes

### *13.4.2.1 Partida 1: Estudios iniciales y planificación inicial*

La partida 1 consiste en la realización de estudios iniciales sobre el proyecto, así como la planificación inicial y la realización de este mismo presupuesto.

La lista de trabajos necesarios es la siguiente:

| **Partida 1: Estudios iniciales y planificación inicial** | | | |
|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***Descripción*** |
| 01 | | | **Estudios iniciales** |
| | 01 | | Definición de los objetivos del proyecto |
| | 02 | | Formación en Sistemas de Recuperación de información y NLP |
| | 03 | | Lectura de papers sobre modelos para la extracción de topics y clasificación de texto |
| | 04 | | Lectura de papers sobre modelos para la generación de resúmenes de textos |
| | 05 | | Búsqueda y selección del dataset |
| | 06 | | Investigación de librerías para la generación de los modelos de aprendizaje automático |
| | 07 | | Comparación y selección de los lenguajes, herramientas, librerías y frameworks a utilizar para implementar el software del proyecto |
| 02 | | | **Planificación y presupuestos iniciales** |
| | 01 | | Planificación inicial |
| | | 01 | Creación de la Estructura de Desglose de Trabajo (WBS) |
| | | 02 | Estimación de actividades |
| | 02 | | Presupuesto inicial |
| | | 01 | Presupuesto de costes inicial |
| | | 02 | Presupuesto de cliente inicial |

#### 13.4.2.1.1 Presupuesto de costes de la Partida 1

| Presupuesto de costes de la partida 1: Estudios iniciales y planificación inicial | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***I4*** | ***Descripción*** | ***Cantidad*** | ***Unidades*** | ***Precio*** | ***Subtotal (4)*** | ***Subtotal (3)*** | ***Subtotal (2)*** | ***Total*** |
| 01 | | | | **Estudios iniciales** | | | | | | | **937,50 €** |
| | 01 | | | Definición de los objetivos del proyecto | | | | | 112,50 € | 112,50 € | |
| | | 01 | | Desarrollador fullstack | 3 | horas | 37,50 € | 112,50 € | | | |
| | 02 | | | Formación en Sistemas de Recuperación de información y NLP | | | | | 225,00 € | 225,00 € | |
| | | 01 | | Desarrollador fullstack | 6 | horas | 37,50 € | 225,00 € | | | |
| | 03 | | | Lectura de papers sobre modelos para la extracción de topics y clasificación de texto | | | | | 300,00 € | 300,00 € | |
| | | 01 | | Desarrollador fullstack | 8 | horas | 37,50 € | 300,00 € | | | |
| | 04 | | | Lectura de papers sobre modelos para la generación de resúmenes de textos | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | 05 | | | Búsqueda y selección del dataset | | | | | 112,50 € | 112,50 € | |
| | | 01 | | Desarrollador fullstack | 3 | horas | 37,50 € | 112,50 € | | | |
| | 06 | | | Investigación de librerías para la generación de los modelos de aprendizaje automático | | | | | 37,50 € | 37,50 € | |
| | | 01 | | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | 07 | | | Comparación y selección de los lenguajes, herramientas, librerías y frameworks a utilizar para implementar el software del proyecto | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| 02 | | | | **Planificación y presupuestos iniciales** | | | | | | | **412,50 €** |
| | 01 | | | Planificación inicial | | | | | | 187,50 € | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 01 | | Creación de la Estructura de Desglose de Trabajo (WBS) | | | | | 150,00 € | | |
| | | | 01 | Desarrollador fullstack | 4 | horas | 37,50 € | 150,00 € | | | |
| | | 02 | | Estimación de actividades | | | | | 37,50 € | | |
| | | | 01 | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | 02 | | | Presupuesto inicial | | | | | | 225,00 € | |
| | | 01 | | Presupuesto de costes inicial | | | | | 150,00 € | | |
| | | | 01 | Desarrollador fullstack | 4 | horas | 37,50 € | 150,00 € | | | |
| | | 02 | | Presupuesto de cliente inicial | | | | | 75,00 € | | |
| | | | 01 | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |

| | |
|---|---|
| **TOTAL** | **1.350,00 €** |

#### 13.4.2.2 Partida 2: Análisis y diseño del sistema

La partida 2 contiene todo lo relativo a las fases de análisis y diseño del sistema.

La lista de trabajos necesarios es la siguiente:

| Partida 2: Análisis y diseño del sistema | | | |
|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***Descripción*** |
| 01 | | | **Análisis** |
| | 01 | | Definición del alcance del sistema |
| | 02 | | Requisitos |
| | | 01 | Obtención de requisitos |
| | | 02 | Especificación de requisitos |
| | 03 | | Casos de Uso |
| | 04 | | Identificación de subsistemas |
| | 05 | | Diagrama de clases preliminar del análisis |
| | 06 | | Análisis de interfaces de usuario |
| | | 01 | Prototipos de pantallas a papel |
| | | 02 | Prototipos de pantallas con herramienta |
| | | 03 | Diagrama de navegabilidad |
| | 07 | | Especificación del plan de pruebas |
| 02 | | | **Diseño** |
| | 01 | | Arquitectura del sistema |
| | | 01 | Diagrama de paquetes |
| | | 02 | Diagrama de componentes |
| | | 03 | Diagramas de despliegue |
| | 02 | | Diseño de clases |
| | | 01 | Identificación de las clases, sus detalles y sus relaciones |
| | | 02 | Diagrama de clases |
| | 03 | | Diseño de la interfaz |
| | 04 | | Especificación técnica del plan de pruebas |

#### 13.4.2.2.1 Presupuesto de costes de la Partida 2

| Presupuesto de costes de la partida 2: Análisis y diseño del sistema | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***I4*** | ***Descripción*** | ***Cantidad*** | ***Unidades*** | ***Precio*** | ***Subtotal (4)*** | ***Subtotal (3)*** | ***Subtotal (2)*** | ***Total*** |
| 01 | | | | **Análisis** | | | | | | | **750,00 €** |
| | 01 | | | Definición del alcance del sistema | | | | | 37,50 € | 37,50 € | |
| | | 01 | | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | 02 | | | Requisitos | | | | | | 150,00 € | |
| | | 01 | | Obtención de requisitos | | | | | 37,50 € | | |
| | | | 01 | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | | 02 | | Especificación de requisitos | | | | | 112,50 € | | |
| | | | 01 | Desarrollador fullstack | 3 | horas | 37,50 € | 112,50 € | | | |
| | 03 | | | Casos de Uso | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | 04 | | | Identificación de subsistemas | | | | | 18,75 € | 18,75 € | |
| | | 01 | | Desarrollador fullstack | 0,5 | horas | 37,50 € | 18,75 € | | | |
| | 05 | | | Diagrama de clases preliminar del análisis | | | | | 225,00 € | 225,00 € | |
| | | 01 | | Desarrollador fullstack | 6 | horas | 37,50 € | 225,00 € | | | |
| | 06 | | | Análisis de interfaces de usuario | | | | | | 206,25 € | |
| | | 01 | | Prototipos de pantallas a papel | | | | | 75,00 € | | |
| | | | 01 | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | | 02 | | Prototipos de pantallas con herramienta | | | | | 112,50 € | | |
| | | | 01 | Desarrollador fullstack | 3 | horas | 37,50 € | 112,50 € | | | |
| | | 03 | | Diagrama de navegabilidad | | | | | 18,75 € | | |
| | | | 01 | Desarrollador fullstack | 0,5 | horas | 37,50 € | 18,75 € | | | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 07 | | | Especificación del plan de pruebas | | | | | 37,50 € | 37,50 € | |
| | | 01 | | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| 02 | | | | **Diseño** | | | | | | | **1.012,50 €** |
| | 01 | | | Arquitectura del sistema | | | | | | 262,50 € | |
| | | 01 | | Diagrama de paquetes | | | | | 112,50 € | | |
| | | | 01 | Desarrollador fullstack | 3 | horas | 37,50 € | 112,50 € | | | |
| | | 02 | | Diagrama de componentes | | | | | 112,50 € | | |
| | | | 01 | Desarrollador fullstack | 3 | horas | 37,50 € | 112,50 € | | | |
| | | 03 | | Diagramas de despliegue | | | | | 37,50 € | | |
| | | | 01 | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | 02 | | | Diseño de clases | | | | | | 450,00 € | |
| | | 01 | | Identificación de las clases, sus detalles y sus relaciones | | | | | 375,00 € | | |
| | | | 01 | Desarrollador fullstack | 10 | horas | 37,50 € | 375,00 € | | | |
| | | 02 | | Diagrama de clases | | | | | 75,00 € | | |
| | | | 01 | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | 03 | | | Diseño de la interfaz | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | 04 | | | Especificación técnica del plan de pruebas | | | | | 225,00 € | 225,00 € | |
| | | 01 | | Desarrollador fullstack | 6 | horas | 37,50 € | 225,00 € | | | |

| | |
|---|---|
| **TOTAL** | **1.762,50 €** |

#### 13.4.2.3 Partida 3: Implementación y despliegue del sistema

La partida 3 contiene todo lo relativo a las fases de análisis y diseño del sistema.

La lista de trabajos necesarios es la siguiente:

| Partida 3: Implementación y despliegue del sistema | | | |
|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***Descripción*** |
| 01 | | | **Implementación** |
| | 01 | | Licencia PyCharm Professional |
| | 02 | | Licencia WebStorm |
| | 03 | | Formación en tecnologías Python y librerías Data science/NLP |
| | 04 | | Implementación del subsistema que contiene los modelos de aprendizaje automático |
| | | 01 | Preparación y preprocesado del dataset |
| | | 02 | Implementación de modelos de aprendizaje automático para la extracción de topics y clasificación de texto |
| | | 03 | Implementación de modelos de aprendizaje automático para la generación de resúmenes de textos |
| | 05 | | Formación en Flask |
| | 06 | | Implementación del subsistema que contiene el backend de la parte web |
| | 07 | | Formación en Angular |
| | 08 | | Implementación del subsistema que contiene el frontend de la parte web |
| 02 | | | **Pruebas** |
| | 01 | | Pruebas unitarias |
| | 02 | | Otras pruebas detalladas en el Plan de Pruebas |
| 03 | | | **Despliegue** |
| | 01 | | Comparar y seleccionar la plataforma de despliegue en la nube |
| | 02 | | Realizar el despliegue en la plataforma seleccionada |

#### 13.4.2.3.1 Presupuesto de costes de la Partida 3

| Presupuesto de costes de la partida 3: Implementación y despliegue del sistema | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***I4*** | ***Descripción*** | ***Cantidad*** | ***Unidades*** | ***Precio*** | ***Subtotal (4)*** | ***Subtotal (3)*** | ***Subtotal (2)*** | ***Total*** |
| 01 | | | | **Implementación** | | | | | | | **4.748,40 €** |
| | 01 | | | Licencia PyCharm Professional | 1 | - | 59,70 € | 59,70 € | 59,70 € | 59,70 € | |
| | 02 | | | Licencia WebStorm | 1 | - | 38,70 € | 38,70 € | 38,70 € | 38,70 € | |
| | 03 | | | Formación en tecnologías Python y librerías Data science/NLP | | | | | 150,00 € | 150,00 € | |
| | | 01 | | Desarrollador fullstack | 4 | horas | 37,50 € | 150,00 € | | | |
| | 04 | | | Implementación del subsistema que contiene los modelos de aprendizaje automático | | | | | | 3.000,00 € | |
| | | 01 | | Preparación y preprocesado del dataset | | | | | 1.125,00 € | | |
| | | | 01 | Desarrollador fullstack | 30 | horas | 37,50 € | 1.125,00 € | | | |
| | | 02 | | Implementación de modelos de aprendizaje automático para la extracción de topics y clasificación de texto | | | | | 1.500,00 € | | |
| | | | 01 | Desarrollador fullstack | 40 | horas | 37,50 € | 1.500,00 € | | | |
| | | 03 | | Implementación de modelos de aprendizaje automático para la generación de resúmenes de textos | | | | | 375,00 € | | |
| | | | 01 | Desarrollador fullstack | 10 | horas | 37,50 € | 375,00 € | | | |
| | 05 | | | Formación en Flask | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | 06 | | | Implementación del subsistema que contiene el backend de la parte web | | | | | 375,00 € | 375,00 € | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 01 | | Desarrollador fullstack | 10 | horas | 37,50 € | 375,00 € | | | |
| | 07 | | | Formación en Angular | | | | | 300,00 € | 300,00 € | |
| | | 01 | | Desarrollador fullstack | 8 | horas | 37,50 € | 300,00 € | | | |
| | 08 | | | Implementación del subsistema que contiene el frontend de la parte web | | | | | 750,00 € | 750,00 € | |
| | | 01 | | Desarrollador fullstack | 20 | horas | 37,50 € | 750,00 € | | | |
| 02 | | | | **Pruebas** | | | | | | | **1.200,00 €** |
| | 01 | | | Pruebas unitarias | | | | | 450,00 € | 450,00 € | |
| | | 01 | | Desarrollador fullstack | 12 | horas | 37,50 € | 450,00 € | | | |
| | 02 | | | Otras pruebas detalladas en el Plan de Pruebas | | | | | 750,00 € | 750,00 € | |
| | | 01 | | Desarrollador fullstack | 20 | horas | 37,50 € | 750,00 € | | | |
| 03 | | | | **Despliegue** | | | | | | | **300,00 €** |
| | 01 | | | Comparar y seleccionar la plataforma de despliegue en la nube | | | | | 37,50 € | 37,50 € | |
| | | 01 | | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | 02 | | | Realizar el despliegue en la plataforma seleccionada | | | | | 262,50 € | 262,50 € | |
| | | 01 | | Desarrollador fullstack | 7 | horas | 37,50 € | 262,50 € | | | |

| | |
|---|---|
| **TOTAL** | **6.248,40 €** |

Dado que la implementación comienza el 10/03/2019 y el despliegue finaliza el 20/05/2019, se ha considerado que las licencias son necesarias durante los meses de marzo, abril y mayo.

| **Cálculo del coste de las licencias** | | | |
|---|---|---|---|
| ***Licencia*** | ***Coste/mes*** | ***Meses de uso*** | ***Coste total*** |
| Licencia PyCharm Professional | 19,90 € | 3 | 59,70 € |
| Licencia WebStorm | 12,90 € | 3 | 38,70 € |

#### 13.4.2.4 Partida 4: Documentación

La partida 4 contiene todo lo relativo a la realización de la documentación del Trabajo Fin de Grado, es decir, el presente documento.

La lista de trabajos necesarios es la siguiente:

| Partida 4: Documentación | | | |
|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***Descripción*** |
| 01 | | | **Introducción** |
| | 01 | | Justificación del proyecto |
| | 02 | | Objetivos del proyecto |
| | 03 | | Estudio de la situación actual |
| 02 | | | **Documentación de la implementación del sistema** |
| | 01 | | Estándares y Normas seguidos |
| | 02 | | Lenguajes de programación |
| | 03 | | Herramientas y Programas usados para el desarrollo |
| | 04 | | Creación del sistema |
| 03 | | | **Documentación del desarrollo de las pruebas** |
| | 01 | | Pruebas unitarias |
| | 02 | | Otras pruebas detalladas en el Plan de Pruebas |
| 04 | | | **Manuales del sistema** |
| | 01 | | Manual de instalación |
| | 02 | | Manual de ejecución |
| | 03 | | Manual de usuario |
| | 04 | | Manual del programador |
| 05 | | | **Aspectos teóricos** |
| | 01 | | Aspectos teóricos |
| 06 | | | **Conclusiones y ampliaciones** |
| | 01 | | Conclusiones |
| | 02 | | Ampliaciones |
| 07 | | | **Memoria del proyecto** |
| | 01 | | Memoria del proyecto |
| 08 | | | **Referencias bibliográficas** |
| | 01 | | Referencias bibliográficas |
| 09 | | | **Apéndices** |
| | 01 | | Glosario |
| | 02 | | Contenido entregado en el archivo adjunto |
| | 03 | | Índice alfabético |
| | 04 | | Código fuente |
| 10 | | | **Agradecimientos, resumen y keywords** |
| | 01 | | Agradecimientos, resumen y keywords |
| 11 | | | **Planificación y presupuestos finales** |
| | 01 | | Planificación final |
| | | 01 | Modificación de la Estructura de Desglose de Trabajo (WBS) |
| | | 02 | Actualización de las estimaciones de las actividades con los tiempos reales |
| | 02 | | Presupuesto final |
| | | 01 | Presupuesto de costes final |
| | | 02 | Presupuesto de cliente final |

#### 13.4.2.4.1 Presupuesto de costes de la Partida 4

| Presupuesto de costes de la partida 4: Documentación | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***I4*** | ***Descripción*** | ***Cantidad*** | ***Unidades*** | ***Precio*** | ***Subtotal (4)*** | ***Subtotal (3)*** | ***Subtotal (2)*** | ***Total*** |
| 01 | | | | **Introducción** | | | | | | | **112,50 €** |
| | 01 | | | Justificación del proyecto | | | | | 37,50 € | 37,50 € | |
| | | 01 | | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | 02 | | | Objetivos del proyecto | | | | | 18,75 € | 18,75 € | |
| | | 01 | | Desarrollador fullstack | 0,5 | horas | 37,50 € | 18,75 € | | | |
| | 03 | | | Estudio de la situación actual | | | | | 56,25 € | 56,25 € | |
| | | 01 | | Desarrollador fullstack | 1,5 | horas | 37,50 € | 56,25 € | | | |
| 02 | | | | **Documentación de la implementación del sistema** | | | | | | | **131,25 €** |
| | 01 | | | Estándares y Normas seguidos | | | | | 18,75 € | 18,75 € | |
| | | 01 | | Desarrollador fullstack | 0,5 | horas | 37,50 € | 18,75 € | | | |
| | 02 | | | Lenguajes de programación | | | | | 18,75 € | 18,75 € | |
| | | 01 | | Desarrollador fullstack | 0,5 | horas | 37,50 € | 18,75 € | | | |
| | 03 | | | Herramientas y Programas usados para el desarrollo | | | | | 37,50 € | 37,50 € | |
| | | 01 | | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | 04 | | | Creación del sistema | | | | | 56,25 € | 56,25 € | |
| | | 01 | | Desarrollador fullstack | 1,5 | horas | 37,50 € | 56,25 € | | | |
| 03 | | | | **Documentación del desarrollo de las pruebas** | | | | | | | **300,00 €** |
| | 01 | | | Pruebas unitarias | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | 02 | | | Otras pruebas detalladas en el Plan de Pruebas | | | | | 225,00 € | 225,00 € | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 01 | | Desarrollador fullstack | 6 | horas | 37,50 € | 225,00 € | | | |
| 04 | | | | **Manuales del sistema** | | | | | | | **431,25 €** |
| | 01 | | | Manual de instalación | | | | | 93,75 € | 93,75 € | |
| | | 01 | | Desarrollador fullstack | 2,5 | horas | 37,50 € | 93,75 € | | | |
| | 02 | | | Manual de ejecución | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | 03 | | | Manual de usuario | | | | | 150,00 € | 150,00 € | |
| | | 01 | | Desarrollador fullstack | 4 | horas | 37,50 € | 150,00 € | | | |
| | 04 | | | Manual del programador | | | | | 112,50 € | 112,50 € | |
| | | 01 | | Desarrollador fullstack | 3 | horas | 37,50 € | 112,50 € | | | |
| 05 | | | | **Aspectos teóricos** | | | | | | | **75,00 €** |
| | 01 | | | Aspectos teóricos | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| 06 | | | | **Conclusiones y ampliaciones** | | | | | | | **131,25 €** |
| | 01 | | | Conclusiones | | | | | 93,75 € | 93,75 € | |
| | | 01 | | Desarrollador fullstack | 2,5 | horas | 37,50 € | 93,75 € | | | |
| | 02 | | | Ampliaciones | | | | | 37,50 € | 37,50 € | |
| | | 01 | | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| 07 | | | | **Memoria del proyecto** | | | | | | | **168,75 €** |
| | 01 | | | Memoria del proyecto | | | | | 168,75 € | 168,75 € | |
| | | 01 | | Desarrollador fullstack | 4,5 | horas | 37,50 € | 168,75 € | | | |
| 08 | | | | **Referencias bibliográficas** | | | | | | | **56,25 €** |
| | 01 | | | Referencias bibliográficas | | | | | 56,25 € | 56,25 € | |
| | | 01 | | Desarrollador fullstack | 1,5 | horas | 37,50 € | 56,25 € | | | |
| 09 | | | | **Apéndices** | | | | | | | **318,75 €** |
| | 01 | | | Glosario | | | | | 56,25 € | 56,25 € | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 01 | | Desarrollador fullstack | 1,5 | horas | 37,50 € | 56,25 € | | | |
| | 02 | | | Contenido entregado en el archivo adjunto | | | | | 131,25 € | 131,25 € | |
| | | 01 | | Desarrollador fullstack | 3,5 | horas | 37,50 € | 131,25 € | | | |
| | 03 | | | Índice alfabético | | | | | 37,50 € | 37,50 € | |
| | | 01 | | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | 04 | | | Código fuente | | | | | 93,75 € | 93,75 € | |
| | | 01 | | Desarrollador fullstack | 2,5 | horas | 37,50 € | 93,75 € | | | |
| 10 | | | | **Agradecimientos, resumen y keywords** | | | | | | | **37,50 €** |
| | 01 | | | Agradecimientos, resumen y keywords | | | | | 37,50 € | 37,50 € | |
| | | 01 | | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| 11 | | | | **Planificación y presupuestos finales** | | | | | | | **225,00 €** |
| | 01 | | | Planificación final | | | | | | 112,50 € | |
| | | 01 | | Modificación de la Estructura de Desglose de Trabajo (WBS) | | | | | 75,00 € | | |
| | | | 01 | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | | 02 | | Actualización de las estimaciones de las actividades con los tiempos reales | | | | | 37,50 € | | |
| | | | 01 | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | 02 | | | Presupuesto final | | | | | | 112,50 € | |
| | | 01 | | Presupuesto de costes final | | | | | 75,00 € | | |
| | | | 01 | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | | 02 | | Presupuesto de cliente final | | | | | 37,50 € | | |
| | | | 01 | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |

| | |
|---|---|
| **TOTAL** | **1.987,50 €** |

### 13.4.2.5 Partida 5: Otros costes

Además de los costes anteriormente identificados, se han identificado lo siguientes costes adicionales: viajes y dietas.

Los **viajes** se corresponden con los traslados del desarrollador a las oficinas de la empresa. Del 01/02/2019 (comienzo del proyecto) al 11/06/2019 (fin del proyecto) hay 93 días laborales (suponemos que durante semana santa no se cogen vacaciones), durante los cuales el desarrollador fullstack se desplazará a las oficinas de la empresa para desarrollar el sistema. Los fines de semana trabajará desde casa. Suponemos que desde la casa del desarrollador a las oficinas de la empresa hay 10km ida y vuelta.

Las **dietas** se corresponden con las comidas que paga la empresa al desarrollador los días laborales que trabaje desde la oficina.

#### 13.4.2.5.1 Presupuesto de costes de la Partida 5

| Presupuesto de costes de la partida 5: Otros costes | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***Descripción*** | ***Cantidad*** | ***Unidades*** | ***Precio*** | ***Subtotal (3)*** | ***Subtotal (2)*** | ***Total*** |
| 01 | | | **Viajes y Dietas** | | | | | | **1.134,60 €** |
| | 01 | | Viajes y Dietas | | | | | 1.134,60 € | |
| | | 01 | Viajes (93 días x 10km/día) | 93 | km | 0,20 € | 18,60 € | | |
| | | 02 | Dietas (93 días) | 93 | dietas | 12,00 € | 1.116,00 € | | |

| | |
|---|---|
| **TOTAL** | **1.134,60 €** |

### 13.4.2.6 Agregación final del presupuesto de costes

Agregando los costes de las cinco partidas se obtiene un resumen final del presupuesto de costes:

| Presupuesto de costes | | |
|---|---|---|
| ***Cód.*** | ***Partida*** | ***Total*** |
| 01 | Estudios iniciales y planificación inicial | 1.350,00 € |
| 02 | Análisis y diseño del sistema | 1.762,50 € |
| 03 | Implementación y despliegue del sistema | 6.248,40 € |
| 04 | Documentación | 1.987,50 € |
| 05 | Otros costes | 1.134,60 € |
| | **Total Coste** | **12.483,00 €** |

## 13.5 Presupuesto de Costes Final (detallado)

El presupuesto de costes final se basa también en el modelo de empresa descrito en el presupuesto de costes inicial, y consta de las mismas partidas.

#### *13.5.1.1 Partida 1: Estudios iniciales y planificación inicial*

La lista de trabajos necesarios es la misma que la de la partida 1 del presupuesto de costes inicial, salvo que en esta lista se elimina el *Presupuesto de cliente inicial*, dado que finalmente no se ha realizado.

#### 13.5.1.1.1 Presupuesto de costes de la Partida 1

| Presupuesto de costes de la partida 1: Estudios iniciales y planificación inicial | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***I4*** | ***Descripción*** | ***Cantidad*** | ***Unidades*** | ***Precio*** | ***Subtotal (4)*** | ***Subtotal (3)*** | ***Subtotal (2)*** | ***Total*** |
| 01 | | | | **Estudios iniciales** | | | | | | | **937,50 €** |
| | 01 | | | Definición de los objetivos del proyecto | | | | | 112,50 € | 112,50 € | |
| | | 01 | | Desarrollador fullstack | 3 | horas | 37,50 € | 112,50 € | | | |
| | 02 | | | Formación en Sistemas de Recuperación de información y NLP | | | | | 225,00 € | 225,00 € | |
| | | 01 | | Desarrollador fullstack | 6 | horas | 37,50 € | 225,00 € | | | |
| | 03 | | | Lectura de papers sobre modelos para la extracción de topics y clasificación de texto | | | | | 300,00 € | 300,00 € | |
| | | 01 | | Desarrollador fullstack | 8 | horas | 37,50 € | 300,00 € | | | |
| | 04 | | | Lectura de papers sobre modelos para la generación de resúmenes de textos | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | 05 | | | Búsqueda y selección del dataset | | | | | 112,50 € | 112,50 € | |
| | | 01 | | Desarrollador fullstack | 3 | horas | 37,50 € | 112,50 € | | | |
| | 06 | | | Investigación de librerías para la generación de los modelos de aprendizaje automático | | | | | 37,50 € | 37,50 € | |
| | | 01 | | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | 07 | | | Comparación y selección de los lenguajes, herramientas, librerías y frameworks a utilizar para implementar el software del proyecto | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| 02 | | | | **Planificación y presupuestos iniciales** | | | | | | | **487,50 €** |
| | 01 | | | Planificación inicial | | | | | | 187,50 € | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 01 | | Creación de la Estructura de Desglose de Trabajo (WBS) | | | | | 150,00 € | | |
| | | | 01 | Desarrollador fullstack | 4 | horas | 37,50 € | 150,00 € | | | |
| | | 02 | | Estimación de actividades | | | | | 37,50 € | | |
| | | | 01 | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | 02 | | | Presupuesto inicial | | | | | | 300,00 € | |
| | | 01 | | Presupuesto de costes inicial | | | | | 300,00 € | | |
| | | | 01 | Desarrollador fullstack | 8 | horas | 37,50 € | 300,00 € | | | |

| | |
|---|---|
| **TOTAL** | **1.425,00 €** |

#### 13.5.1.2 Partida 2: Análisis y diseño del sistema

La lista de trabajos necesarios es la siguiente:

| Partida 2: Análisis y diseño del sistema | | | |
|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***Descripción*** |
| 01 | | | **Análisis** |
| | 01 | | Definición del alcance del sistema |
| | 02 | | Requisitos |
| | | 01 | Obtención de requisitos |
| | | 02 | Especificación de requisitos |
| | 03 | | Casos de Uso |
| | 04 | | Identificación de subsistemas |
| | 05 | | Diagrama de clases preliminar del análisis |
| | 06 | | Análisis de interfaces de usuario |
| | | 01 | Prototipos de pantallas a papel |
| | | 02 | Pasar los prototipos de pantallas en papel a una herramienta |
| | 07 | | Especificación del plan de pruebas |
| 02 | | | **Diseño** |
| | 01 | | Arquitectura del sistema |
| | | 01 | Diagramas de componentes |
| | | 02 | Diagramas de despliegue |
| | 02 | | Diseño de clases |
| | | 01 | Identificación de las clases, sus detalles y sus relaciones |
| | | 02 | Diagramas de clases |
| | 03 | | Diseño de la interfaz |
| | | 01 | Prototipo interactivo de las pantallas con FluidUI |
| | 04 | | Especificación técnica del plan de pruebas |

#### 13.5.1.2.1 Presupuesto de costes de la Partida 2

| Presupuesto de costes de la partida 2: Análisis y diseño del sistema | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***I4*** | ***Descripción*** | ***Cantidad*** | ***Unidades*** | ***Precio*** | ***Subtotal (4)*** | ***Subtotal (3)*** | ***Subtotal (2)*** | ***Total*** |
| 01 | | | | **Análisis** | | | | | | | **937,50 €** |
| | 01 | | | Definición del alcance del sistema | | | | | 18,75 € | 18,75 € | |
| | | 01 | | Desarrollador fullstack | 0,5 | horas | 37,50 € | 18,75 € | | | |
| | 02 | | | Requisitos | | | | | | 318,75 € | |
| | | 01 | | Obtención de requisitos | | | | | 131,25 € | | |
| | | | 01 | Desarrollador fullstack | 3,5 | horas | 37,50 € | 131,25 € | | | |
| | | 02 | | Especificación de requisitos | | | | | 187,50 € | | |
| | | | 01 | Desarrollador fullstack | 5 | horas | 37,50 € | 187,50 € | | | |
| | 03 | | | Casos de Uso | | | | | 112,50 € | 112,50 € | |
| | | 01 | | Desarrollador fullstack | 3 | horas | 37,50 € | 112,50 € | | | |
| | 04 | | | Identificación de subsistemas | | | | | 18,75 € | 18,75 € | |
| | | 01 | | Desarrollador fullstack | 0,5 | horas | 37,50 € | 18,75 € | | | |
| | 05 | | | Diagrama de clases preliminar del análisis | | | | | 225,00 € | 225,00 € | |
| | | 01 | | Desarrollador fullstack | 6 | horas | 37,50 € | 225,00 € | | | |
| | 06 | | | Análisis de interfaces de usuario | | | | | | 206,25 € | |
| | | 01 | | Prototipos de pantallas a papel | | | | | 150,00 € | | |
| | | | 01 | Desarrollador fullstack | 4 | horas | 37,50 € | 150,00 € | | | |
| | | 02 | | Pasar los prototipos de pantallas en papel a una herramienta | | | | | 56,25 € | | |
| | | | 01 | Desarrollador fullstack | 1,5 | horas | 37,50 € | 56,25 € | | | |
| | 07 | | | Especificación del plan de pruebas | | | | | 37,50 € | 37,50 € | |
| | | 01 | | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 02 | | | | **Diseño** | | | | | | | **1.368,75 €** |
| | 01 | | | Arquitectura del sistema | | | | | | 225,00 € | |
| | | 01 | | Diagramas de componentes | | | | | 150,00 € | | |
| | | | 01 | Desarrollador fullstack | 4 | horas | 37,50 € | 150,00 € | | | |
| | | 02 | | Diagramas de despliegue | | | | | 75,00 € | | |
| | | | 01 | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | 02 | | | Diseño de clases | | | | | | 731,25 € | |
| | | 01 | | Identificación de las clases, sus detalles y sus relaciones | | | | | 375,00 € | | |
| | | | 01 | Desarrollador fullstack | 10 | horas | 37,50 € | 375,00 € | | | |
| | | 02 | | Diagramas de clases | | | | | 356,25 € | | |
| | | | 01 | Desarrollador fullstack | 9,5 | horas | 37,50 € | 356,25 € | | | |
| | 03 | | | Diseño de la interfaz | | | | | 225,00 € | 225,00 € | |
| | | 01 | | Prototipo interactivo de las pantallas con FluidUI | | | | | | | |
| | | | 01 | Desarrollador fullstack | 6 | horas | 37,50 € | 225,00 € | | | |
| | 04 | | | Especificación técnica del plan de pruebas | | | | | 187,50 € | 187,50 € | |
| | | 01 | | Desarrollador fullstack | 5 | horas | 37,50 € | 187,50 € | | | |

| | |
|---|---|
| **TOTAL** | **2.306,25 €** |

#### 13.5.1.3 Partida 3: Implementación y despliegue del sistema

| Partida 3: Implementación y despliegue del sistema | | | |
|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***Descripción*** |
| 01 | | | **Implementación** |
| | 01 | | Licencia PyCharm Professional |
| | 02 | | Licencia WebStorm |
| | 03 | | Formación en tecnologías Python y librerías Data science/NLP |
| | 04 | | **Implementación del subsistema "Topics and Summary"** |
| | | 01 | Componente datasets |
| | | 02 | Componente preprocessing |
| | | 03 | Componente embeddings |
| | | 04 | Componente models |
| | | 05 | Componente visualizations |
| | | 06 | Demo con la funcionalidad del subsistema |
| | | 07 | Transformación del subsistema en una librería Python instalable |
| | | 08 | Implementación de los ficheros Dockerfile y Vagrantfile |
| | 05 | | Pruebas unitarias del subsistema "Topics and Summary" |
| | 06 | | Generación y comparativa de varios modelos de extracción de topics, y selección del modelo final |
| | 07 | | Formación en Flask |
| | 08 | | **Implementación del subsistema "Web backend"** |
| | | 01 | Componente params |
| | | 02 | Componente wrapper |
| | | 03 | Componente apis.user |
| | | 04 | Implementación del fichero Dockerfile |
| | | 05 | Despliegue del subsistema |
| | 09 | | Pruebas unitarias del subsistema "Web backend" |
| | 10 | | Formación en Angular |
| | 11 | | **Implementación del subsistema "Web frontend"** |
| | | 01 | Componente topics |
| | | 02 | Componente text |
| | | 03 | Componente app |
| | | 04 | Componente header |
| | | 05 | Componente footer |
| | | 06 | Implementación del fichero Dockerfile |
| | | 07 | Despliegue del subsistema |
| 02 | | | **Pruebas** |
| | 01 | | Pruebas de usabilidad |
| | | 01 | Diseñar los cuestionarios |
| | | 02 | Realizar las pruebas de usabilidad a los 3 usuarios |
| | | 03 | Analizar los resultados y sacar conclusiones |
| | 02 | | Pruebas de adaptabilidad |
| | 03 | | Análisis estático de código |

#### 13.5.1.3.1 Presupuesto de costes de la Partida 3

| Presupuesto de costes de la partida 3: Implementación y despliegue del sistema | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***I4*** | ***Descripción*** | ***Cantidad*** | ***Unidades*** | ***Precio*** | ***Subtotal (4)*** | ***Subtotal (3)*** | ***Subtotal (2)*** | ***Total*** |
| 01 | | | | **Implementación** | | | | | | | **9.824,95 €** |
| | 01 | | | Licencia PyCharm Professional | 1 | - | 79,60 € | 79,60 € | 79,60 € | 79,60 € | |
| | 02 | | | Licencia WebStorm | 1 | - | 51,60 € | 51,60 € | 51,60 € | 51,60 € | |
| | 03 | | | Formación en tecnologías Python y librerías Data science/NLP | | | | | 150,00 € | 150,00 € | |
| | | 01 | | Desarrollador fullstack | 4 | horas | 37,50 € | 150,00 € | | | |
| | 04 | | | **Implementación del subsistema "Topics and Summary"** | | | | | | 4.575,00 € | |
| | | 01 | | Componente datasets | | | | | 562,50 € | | |
| | | | 01 | Desarrollador fullstack | 15 | horas | 37,50 € | 562,50 € | | | |
| | | 02 | | Componente preprocessing | | | | | 1.312,50 € | | |
| | | | 01 | Desarrollador fullstack | 35 | horas | 37,50 € | 1.312,50 € | | | |
| | | 03 | | Componente embeddings | | | | | 112,50 € | | |
| | | | 01 | Desarrollador fullstack | 3 | horas | 37,50 € | 112,50 € | | | |
| | | 04 | | Componente models | | | | | 1.650,00 € | | |
| | | | 01 | Desarrollador fullstack | 44 | horas | 37,50 € | 1.650,00 € | | | |
| | | 05 | | Componente visualizations | | | | | 375,00 € | | |
| | | | 01 | Desarrollador fullstack | 10 | horas | 37,50 € | 375,00 € | | | |
| | | 06 | | Demo con la funcionalidad del subsistema | | | | | 187,50 € | | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | 01 | Desarrollador fullstack | 5 | horas | 37,50 € | 187,50 € | | | |
| | | 07 | | Transformación del subsistema en una librería Python instalable | | | | | 150,00 € | | |
| | | | 01 | Desarrollador fullstack | 4 | horas | 37,50 € | 150,00 € | | | |
| | | 08 | | Implementación de los ficheros Dockerfile y Vagrantfile | | | | | 225,00 € | | |
| | | | 01 | Desarrollador fullstack | 6 | horas | 37,50 € | 225,00 € | | | |
| | 05 | | | Pruebas unitarias del subsistema "Topics and Summary" | | | | | 300,00 € | 300,00 € | |
| | | 01 | | Desarrollador fullstack | 8 | horas | 37,50 € | 300,00 € | | | |
| | 06 | | | Generación y comparativa de varios modelos de extracción de topics, y selección del modelo final | | | | | 525,00 € | 525,00 € | |
| | | 01 | | Desarrollador fullstack | 14 | horas | 37,50 € | 525,00 € | | | |
| | 07 | | | Formación en Flask | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | 08 | | | **Implementación del subsistema "Web backend"** | | | | | | 975,00 € | |
| | | 01 | | Componente params | | | | | 93,75 € | | |
| | | | 01 | Desarrollador fullstack | 2,5 | horas | 37,50 € | 93,75 € | | | |
| | | 02 | | Componente wrapper | | | | | 318,75 € | | |
| | | | 01 | Desarrollador fullstack | 8,5 | horas | 37,50 € | 318,75 € | | | |
| | | 03 | | Componente apis.user | | | | | 262,50 € | | |
| | | | 01 | Desarrollador fullstack | 7 | horas | 37,50 € | 262,50 € | | | |
| | | 04 | | Implementación del fichero Dockerfile | | | | | 225,00 € | | |
| | | | 01 | Desarrollador fullstack | 6 | horas | 37,50 € | 225,00 € | | | |
| | | 05 | | Despliegue del subsistema | | | | | 75,00 € | | |
| | | | 01 | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | 09 | | | Pruebas unitarias del subsistema "Web backend" | | | | | 75,00 € | 75,00 € | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | 10 | | | Formación en Angular | | | | | 300,00 € | 300,00 € | |
| | | 01 | | Desarrollador fullstack | 8 | horas | 37,50 € | 300,00 € | | | |
| | 11 | | | **Implementación del subsistema "Web frontend"** | | | | | | 2.718,75 € | |
| | | 01 | | Componente topics | | | | | 1.125,00 € | | |
| | | | 01 | Desarrollador fullstack | 30 | horas | 37,50 € | 1.125,00 € | | | |
| | | 02 | | Componente text | | | | | 1.312,50 € | | |
| | | | 01 | Desarrollador fullstack | 35 | horas | 37,50 € | 1.312,50 € | | | |
| | | 03 | | Componente app | | | | | 37,50 € | | |
| | | | 01 | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | | 04 | | Componente header | | | | | 37,50 € | | |
| | | | 01 | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | | 05 | | Componente footer | | | | | 18,75 € | | |
| | | | 01 | Desarrollador fullstack | 0,5 | horas | 37,50 € | 18,75 € | | | |
| | | 06 | | Implementación del fichero Dockerfile | | | | | 150,00 € | | |
| | | | 01 | Desarrollador fullstack | 4 | horas | 37,50 € | 150,00 € | | | |
| | | 07 | | Despliegue del subsistema | | | | | 37,50 € | | |
| | | | 01 | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| 02 | | | | **Pruebas** | | | | | | | **487,50 €** |
| | 01 | | | Pruebas de usabilidad | | | | | | 187,50 € | |
| | | 01 | | Diseñar los cuestionarios | | | | | 37,50 € | | |
| | | | 01 | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | | 02 | | Realizar las pruebas de usabilidad a los 3 usuarios | | | | | 112,50 € | | |
| | | | 01 | Desarrollador fullstack | 3 | horas | 37,50 € | 112,50 € | | | |
| | | 03 | | Analizar los resultados y sacar conclusiones | | | | | 37,50 € | | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | 01 | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | 02 | | | Pruebas de adaptabilidad | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | 03 | | | Análisis estático de código | | | | | 225,00 € | 225,00 € | |
| | | 01 | | Desarrollador fullstack | 6 | horas | 37,50 € | 225,00 € | | | |

| | |
|---|---|
| **TOTAL** | **10.312,45 €** |

Dado que la implementación comenzó el 09/03/2019 y la implementación de las pruebas finalizó el 11/06/2019, se va a considerar que las licencias fueron necesarias durante los meses de marzo, abril, mayo y junio.

| **Cálculo del coste de las licencias** | | | |
|---|---|---|---|
| ***Licencia*** | ***Coste/mes*** | ***Meses de uso*** | ***Coste total*** |
| Licencia PyCharm Professional | 19,90 € | 4 | 79,60 € |
| Licencia WebStorm | 12,90 € | 4 | 51,60 € |

#### 13.5.1.4 Partida 4: Documentación

La lista de trabajos necesarios es la siguiente:

| Partida 4: Documentación | | | |
|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***Descripción*** |
| 01 | | | **Introducción** |
| | 01 | | Justificación del proyecto |
| | 02 | | Objetivos del proyecto |
| | 03 | | Estudio de la situación actual |
| 02 | | | **Documentación de la implementación del sistema** |
| | 01 | | Estándares y Normas seguidos |
| | 02 | | Lenguajes de programación |
| | 03 | | Herramientas y Programas usados para el desarrollo |
| | 04 | | Creación del sistema |
| 03 | | | **Documentación del desarrollo de las pruebas** |
| | 01 | | Pruebas unitarias automatizadas |
| 04 | | | **Manuales del sistema** |
| | 01 | | Documentación del subsistema "Topics and Summary" |
| | 02 | | Documentación del subsistema "Web backend" |
| | 03 | | Documentación de la API REST con Swagger |
| | 04 | | Documentación del subsistema "Web frontend" |
| 05 | | | **Aspectos teóricos** |
| | 01 | | Aspectos teóricos |
| 06 | | | **Conclusiones y ampliaciones** |
| | 01 | | Conclusiones |
| | 02 | | Ampliaciones |
| 07 | | | **Memoria del proyecto** |
| | 01 | | Memoria del proyecto |
| 08 | | | **Referencias bibliográficas** |
| | 01 | | Referencias bibliográficas |
| 09 | | | **Apéndices** |
| | 01 | | Glosario |
| | 02 | | Contenido entregado en el archivo adjunto |
| | 03 | | Código fuente |
| | 04 | | Documentación del presupuesto de costes inicial (detallado) |
| | 05 | | Documentación del presupuesto de costes final (detallado) |
| 10 | | | **Agradecimientos, resumen y keywords** |
| | 01 | | Agradecimientos, resumen y keywords |
| 11 | | | **Planificación y presupuestos finales** |
| | 01 | | Planificación final |
| | | 01 | Modificación de la Estructura de Desglose de Trabajo (WBS) |
| | | 02 | Actualización de las estimaciones de las actividades con los tiempos reales |
| | 02 | | Presupuesto final |
| | | 01 | Presupuesto de costes final |

#### 13.5.1.4.1 Presupuesto de costes de la Partida 4

| Presupuesto de costes de la partida 4: Documentación | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***I4*** | ***Descripción*** | ***Cantidad*** | ***Unidades*** | ***Precio*** | ***Subtotal (4)*** | ***Subtotal (3)*** | ***Subtotal (2)*** | ***Total*** |
| 01 | | | | **Introducción** | | | | | | | **187,50 €** |
| | 01 | | | Justificación del proyecto | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | 02 | | | Objetivos del proyecto | | | | | 18,75 € | 18,75 € | |
| | | 01 | | Desarrollador fullstack | 0,5 | horas | 37,50 € | 18,75 € | | | |
| | 03 | | | Estudio de la situación actual | | | | | 93,75 € | 93,75 € | |
| | | 01 | | Desarrollador fullstack | 2,5 | horas | 37,50 € | 93,75 € | | | |
| 02 | | | | **Documentación de la implementación del sistema** | | | | | | | **150,00 €** |
| | 01 | | | Estándares y Normas seguidos | | | | | 9,38 € | 9,38 € | |
| | | 01 | | Desarrollador fullstack | 0,25 | horas | 37,50 € | 9,38 € | | | |
| | 02 | | | Lenguajes de programación | | | | | 9,38 € | 9,38 € | |
| | | 01 | | Desarrollador fullstack | 0,25 | horas | 37,50 € | 9,38 € | | | |
| | 03 | | | Herramientas y Programas usados para el desarrollo | | | | | 18,75 € | 18,75 € | |
| | | 01 | | Desarrollador fullstack | 0,5 | horas | 37,50 € | 18,75 € | | | |
| | 04 | | | Creación del sistema | | | | | 112,50 € | 112,50 € | |
| | | 01 | | Desarrollador fullstack | 3 | horas | 37,50 € | 112,50 € | | | |
| 03 | | | | **Documentación del desarrollo de las pruebas** | | | | | | | **75,00 €** |
| | 01 | | | Pruebas unitarias automatizadas | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| 04 | | | | **Manuales del sistema** | | | | | | | **1.537,50 €** |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 01 | | | Documentación del subsistema "Topics and Summary" | | | | | 750,00 € | 750,00 € | |
| | | 01 | | Desarrollador fullstack | 20 | horas | 37,50 € | 750,00 € | | | |
| | 02 | | | Documentación del subsistema "Web backend" | | | | | 375,00 € | 375,00 € | |
| | | 01 | | Desarrollador fullstack | 10 | horas | 37,50 € | 375,00 € | | | |
| | 03 | | | Documentación de la API REST con Swagger | | | | | 112,50 € | 112,50 € | |
| | | 01 | | Desarrollador fullstack | 3 | horas | 37,50 € | 112,50 € | | | |
| | 04 | | | Documentación del subsistema "Web frontend" | | | | | 300,00 € | 300,00 € | |
| | | 01 | | Desarrollador fullstack | 8 | horas | 37,50 € | 300,00 € | | | |
| 05 | | | | **Aspectos teóricos** | | | | | | | **150,00 €** |
| | 01 | | | Aspectos teóricos | | | | | 150,00 € | 150,00 € | |
| | | 01 | | Desarrollador fullstack | 4 | horas | 37,50 € | 150,00 € | | | |
| 06 | | | | **Conclusiones y ampliaciones** | | | | | | | **131,25 €** |
| | 01 | | | Conclusiones | | | | | 56,25 € | 56,25 € | |
| | | 01 | | Desarrollador fullstack | 1,5 | horas | 37,50 € | 56,25 € | | | |
| | 02 | | | Ampliaciones | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| 07 | | | | **Memoria del proyecto** | | | | | | | **112,50 €** |
| | 01 | | | Memoria del proyecto | | | | | 112,50 € | 112,50 € | |
| | | 01 | | Desarrollador fullstack | 3 | horas | 37,50 € | 112,50 € | | | |
| 08 | | | | **Referencias bibliográficas** | | | | | | | **75,00 €** |
| | 01 | | | Referencias bibliográficas | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| 09 | | | | **Apéndices** | | | | | | | **281,25 €** |
| | 01 | | | Glosario | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | 02 | | | Contenido entregado en el archivo adjunto | | | | | 9,38 € | 9,38 € | |
| | | 01 | | Desarrollador fullstack | 0,25 | horas | 37,50 € | 9,38 € | | | |
| | 03 | | | Código fuente | | | | | 9,38 € | 9,38 € | |
| | | 01 | | Desarrollador fullstack | 0,25 | horas | 37,50 € | 9,38 € | | | |
| | 04 | | | Documentación del presupuesto de costes inicial (detallado) | | | | | 150,00 € | 150,00 € | |
| | | 01 | | Desarrollador fullstack | 4 | horas | 37,50 € | 150,00 € | | | |
| | 05 | | | Documentación del presupuesto de costes final (detallado) | | | | | 37,50 € | 37,50 € | |
| | | 01 | | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| 10 | | | | **Agradecimientos, resumen y keywords** | | | | | | | **75,00 €** |
| | 01 | | | Agradecimientos, resumen y keywords | | | | | 75,00 € | 75,00 € | |
| | | 01 | | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| 11 | | | | **Planificación y presupuestos finales** | | | | | | | **187,50 €** |
| | 01 | | | Planificación final | | | | | | 112,50 € | |
| | | 01 | | Modificación de la Estructura de Desglose de Trabajo (WBS) | | | | | 75,00 € | | |
| | | | 01 | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |
| | | 02 | | Actualización de las estimaciones de las actividades con los tiempos reales | | | | | 37,50 € | | |
| | | | 01 | Desarrollador fullstack | 1 | horas | 37,50 € | 37,50 € | | | |
| | 02 | | | Presupuesto final | | | | | | 75,00 € | |
| | | 01 | | Presupuesto de costes final | | | | | 75,00 € | | |
| | | | 01 | Desarrollador fullstack | 2 | horas | 37,50 € | 75,00 € | | | |

| | |
|---|---|
| **TOTAL** | **2.962,50 €** |

### 13.5.1.5 Partida 5: Otros costes

Además de los costes anteriormente identificados, se mantienen como costes adicionales los viajes y dietas, al igual que en la planificación inicial.

Los **viajes** se corresponden con los traslados del desarrollador a las oficinas de la empresa. Del 01/02/2019 (comienzo del proyecto) al 07/07/2019 (fin del proyecto) hay 111 días laborales (suponemos que durante semana santa no se cogen vacaciones), durante los cuales el desarrollador fullstack se desplazará a las oficinas de la empresa para desarrollar el sistema. Los fines de semana trabajará desde casa. Suponemos que desde la casa del desarrollador a las oficinas de la empresa hay 10km ida y vuelta.

Las **dietas** se corresponden con las comidas que paga la empresa al desarrollador los días laborales que trabaje desde la oficina.

#### 13.5.1.5.1 Presupuesto de costes de la Partida 5

| Presupuesto de costes de la partida 5: Otros costes | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| ***I1*** | ***I2*** | ***I3*** | ***Descripción*** | ***Cantidad*** | ***Unidades*** | ***Precio*** | ***Subtotal (3)*** | ***Subtotal (2)*** | ***Total*** |
| 01 | | | **Viajes y Dietas** | | | | | | **1.354,20 €** |
| | 01 | | Viajes y Dietas | | | | | 1.354,20 € | |
| | | 01 | Viajes (111 días x 10km/día) | 111 | km | 0,20 € | 22,20 € | | |
| | | 02 | Dietas (111 días) | 111 | dietas | 12,00 € | 1.332,00 € | | |

| | |
|---|---|
| **TOTAL** | **1.354,20 €** |

### 13.5.1.6 Agregación final del presupuesto de costes

Agregando los costes de las cinco partidas se obtiene un resumen final del presupuesto de costes:

| Presupuesto de costes | | |
|---|---|---|
| ***Cód.*** | ***Partida*** | ***Total*** |
| 01 | Estudios iniciales y planificación inicial | 1.425,00 € |
| 02 | Análisis y diseño del sistema | 2.306,25 € |
| 03 | Implementación y despliegue del sistema | 10.312,45 € |
| 04 | Documentación | 2.962,50 € |
| 05 | Otros costes | 1.354,20 € |
| | **Total Coste** | **18.360,40 €** |