



UNIVERSIDAD DE JAÉN
Escuela Politécnica Superior

Trabajo Fin de Grado

PLATAFORMA DE ANÁLISIS DE LA COMPLEJIDAD DE TEXTOS EN ESPAÑOL

Alumno: José Manuel Rocha Ruiz

Tutor: Arturo Montejo Ráez
Dpto: Departamento de informática

Junio, 2018



Universidad de Jaén
Escuela Politécnica Superior de Jaén
Departamento de Informática

Don ARTURO MONTEJO RAEZ, tutor del Proyecto Fin de Carrera titulado:
PLATAFORMA DE ANÁLISIS DE LA COMPLEJIDAD DE TEXTOS EN
ESPAÑOL, que presenta JOSÉ MANUEL ROCHA RUIZ, autoriza su
presentación para defensa y evaluación en la Escuela Politécnica Superior de
Jaén.

Jaén, JUNIO de 2018

El alumno:

Los tutores:

JOSÉ MANUEL ROCHA RUIZ

ARTURO MONTEJO RAEZ

TABLA DE CONTENIDOS

1. Definición del problema – Objetivos	4
2. Marco de referencia.....	4
2.1 Aplicación web	4
2.2 Procesamiento del lenguaje natural	5
3. Metodología	5
4. Requisitos funcionales	6
5. Análisis y diseño	7
5.1. Análisis	7
5.1.1 Casos de uso	8
5.1.2 Diagramas de secuencia.....	17
5.2 Diseño	21
5.2.1 Mockup.....	22
6. Tecnologías y lenguajes	28
6.1 Pyhton.....	29
6.2 PyPdf2.....	29
6.3 MongoDB.....	30
6.4 XML.....	30
6.5 Freeling	31
6.6 Flask	31
6.7 HTML	32
6.8 Bootstrap	32
6.9 Fontawesome.....	32
6.10 CSS.....	33
6.11 Javascript	33
6.12 Google Charts.....	34
8. Discusión	35
9. Trabajo futuro	35
10. Bibliografía	37
11. Anexo: Manual de despliegue	38

1. Definición del problema – Objetivos

Se desea elaborar una aplicación web que permita a sus usuarios saber la dificultad léxica y sintáctica de los textos que se faciliten, para ello se desean calcular distintas medidas de complejidad del mismo. Dicha aplicación irá orientada al ámbito editorial y educativo. Para ello, se plantean los siguientes objetivos:

- Elaborar una herramienta sencilla de para la obtención de indicadores de complejidad del texto.
- Profundizar en la clasificación automática de textos en base a su complejidad.
- Implementar un sistema web completo, con una base de datos y usuarios con distintos niveles de privilegios.
- Dados los requisitos del problema, se utilizará una base de datos no relacional.

2. Marco de referencia

A continuación, se describen los distintos conceptos que ayudarán a contextualizar el proyecto y entender mejor el desarrollo.

2.1 Aplicación web

Las aplicaciones web son un tipo de programas desarrollados en lenguajes entendibles por los navegadores web y cuya ejecución es llevada a cabo por el navegador en Internet. Gracias a que las aplicaciones web son ejecutadas mediante un navegador web, los datos manipulados son procesados

y almacenados en una base de datos ubicada en el servidor. Por este motivo, este tipo de aplicaciones no necesitan ser instaladas en el dispositivo, permitiendo una gran portabilidad.

La finalidad de las aplicaciones web, a diferencia de un sitio web, es que la utilidad de las mismas radica en que ayudan al usuario a realizar tareas, mientras que un sitio web es importante fundamentalmente por la información que muestra.

2.2 Procesamiento del lenguaje natural

El Procesamiento del Lenguaje Natural es un área de investigación y aplicación que explora cómo utilizar los equipos informáticos para entender y manipular textos o discursos en lenguaje natural con el fin de obtener información útil. Los investigadores de PLN intentan recoger información sobre cómo los seres humanos entendemos y utilizamos el lenguaje para así poder desarrollar técnicas apropiadas para que los sistemas computacionales puedan entender y manipular el lenguaje natural.

Los fundamentos del PNL se basan en una serie de campos científicos, como son la informática, filología, matemáticas, inteligencia artificial y robótica, psicología, etc. Las aplicaciones de PNL incluyen una serie de campos de estudio, como traducción automática, procesamiento y resumen de texto en lenguaje natural, interfaces de usuario y recuperación de información de lenguaje cruzado (CLIR), reconocimiento de voz, inteligencia artificial y sistemas expertos.
[1]

3. Metodología

Para realizar el desarrollo del proyecto se ha llevado a cabo la siguiente metodología

- Análisis: Especificación de las necesidades y objetivos que debe cubrir la aplicación.
- Diseño: Definición visual del sistema. Formulación de especificaciones para el nuevo sistema, de forma que satisfaga los requisitos determinados durante la fase de análisis. Al final se obtiene un modelo detallado listo para implementar.
- Desarrollo: Trasladar el diseño a un modelo funcional mediante lenguajes de programación
- Pruebas: Ejecutar y depurar la aplicación en busca de errores.
- Redacción de la memoria: Estructurar y escribir en un documento las anotaciones recolectadas durante todo el desarrollo de las fases anteriores

4. Requisitos funcionales

- La aplicación obliga registrarse a nuevos usuarios para poder utilizarla
- La aplicación debe distinguir entre usuarios básicos, usuarios Premium y usuarios administradores
 - Los usuarios básicos solamente pueden utilizar la aplicación de forma restringida, muestra únicamente las medidas de complejidad básicas de los documentos facilitados. Los usuarios básicos sólo podrán almacenar un número limitado de documentos.
 - Los usuarios Premium tendrán acceso a todas las medidas de complejidad, así como a los valores utilizados para calcularlas y gráficas comparativas.

- Los usuarios administradores tendrán acceso al backend de la aplicación, de forma que podrán manipular los datos de cualquier usuario, a excepción de la contraseña. También tendrán la posibilidad de manipular los documentos almacenados en el servidor.
- La aplicación debe ser responsive, adaptada para su uso en dispositivos con diferentes resoluciones de pantalla.

5. Análisis y diseño

Esta sección se dedica a la especificación del proyecto, reuniendo los requisitos y características que deberá tener el mismo.

5.1. Análisis

En la etapa de análisis se ha procedido a la creación de distintos casos de uso, que muestran en lenguaje normal las distintas situaciones que puedan darse durante el uso de la aplicación. Mediante los casos de uso se muestran los contextos de desarrollo de interacción con la aplicación y las distintas secuencias de eventos.

Posteriormente se generan los diagramas de secuencia, los cuales muestran las interacciones entre las distintas clases de la aplicación a nivel interno, las respuestas producidas, los ciclos de vida de las distintas clases y las alternativas propuestas.

5.1.1 Casos de uso

Diagrama de casos de uso aplicación web

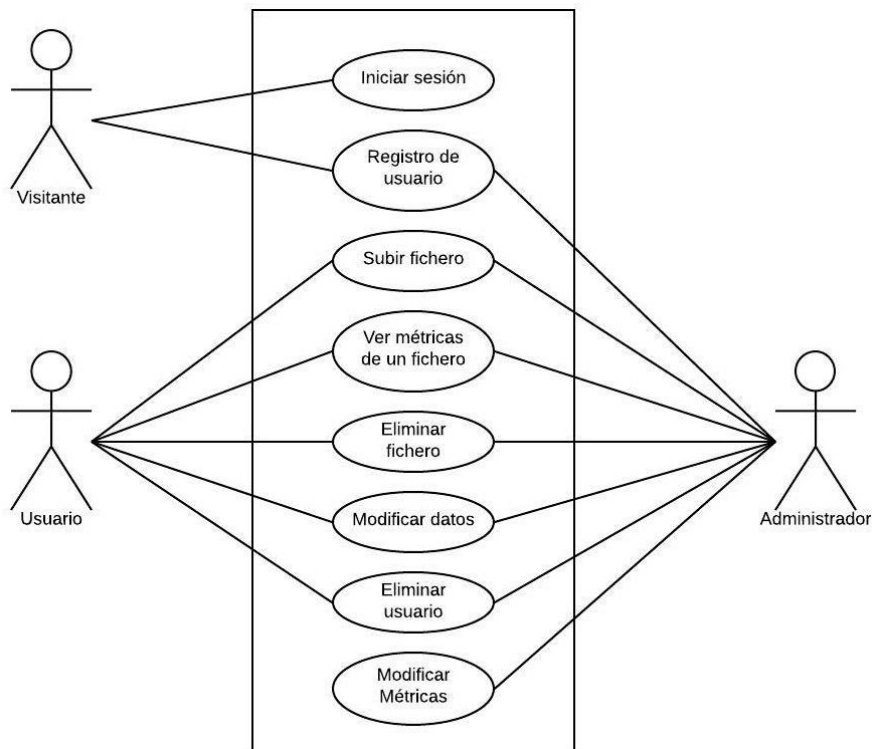


Ilustración 1 Diagrama de casos de uso aplicación web

Especificación del caso de uso

CU1: Iniciar sesión

Descripción: Permite a un visitante acceder a la aplicación con algunos o todos los permisos.

Actores: Visitante.

Precondiciones:

- El visitante tiene que estar registrado en el sistema.

Operaciones básicas	Actor	Respuesta del sistema
1.	Solicita el acceso a la aplicación	La aplicación muestra un formulario de validación de datos
2.	Ingresa su nombre de usuario y contraseña	
Alternativas		
2.A		¿Son correctos los datos?
2.A.1		Si sí, continuar, el actor pasa a tener rol de usuario o administrador
2.A.2		Si no, volver a 1

Tabla 1. Flujo de eventos - Iniciar sesión

Especificación del caso de uso**CU2:** Registro de usuario.

Descripción: Permite a un visitante no registrado en la aplicación que se le registre, almacenando sus datos en la base de datos de la aplicación.

Actores: Visitante.

Precondiciones: Ninguna.

Operaciones básicas	Actor	Respuesta del sistema
1.	Pulsa sobre el botón para acceder al formulario de registro en la aplicación	Muestra el formulario para introducir los datos básicos de usuario: nombre,

		apellidos, email, contraseña, username.
2.	Introduce en el formulario los datos solicitados	
3.	El visitante pulsa sobre el botón para registrarse en la aplicación	La aplicación comprueba la validez de los datos introducidos y los almacena en la base de datos.
Alternativas		
3.A		Comprueba que todos los campos están completos.
3.A.1		Si sí, continúa
3.A.2		Si no, avisa al actor y vuelve a 2
3.B		Comprueba que las contraseñas introducidas sean iguales
3.B.1		Si sí, continúa
3.B.2		Si no, vuelve a 2, avisa al actor de que son diferentes
3.C		Comprueba que el username no existe
3.C.1		Si sí, vuelve a 2, avisa al actor de que ya existe, permite modificarlo
3.C.2		Si no, continuar, el actor pasa a estar registrado y accede a la aplicación con el rol correspondiente.

Tabla 2. Flujo de eventos - Registro de usuario

Especificación del caso de uso**CU3:** Subir fichero.

Descripción: Permite subir un documento “.pdf” o “.txt” al servidor e indexarlo en la base de datos, para que la aplicación lo analice y muestre posteriormente las medidas resultantes de dicho análisis.

Actores: Usuario, Administrador.

Precondiciones:

- El fichero debe ser en formato “.pdf” o “.txt”

Operaciones básicas	Actor	Respuesta del sistema
1.	Selecciona los ficheros locales a subir	Muestra la cantidad de ficheros seleccionados
2.	Pulsa para comenzar la subida de los ficheros	Muestra el porcentaje de subida
Alternativas		
2.A		¿Ha subido todos los ficheros?
2.A.1		Si sí, indica que se han subido correctamente
2.A.2		Si no, indica que no se han podido subir

Tabla 3. Flujo de eventos - subir fichero

Especificación del caso de uso**CU4:** Ver métricas de un fichero**Descripción:** Cuando el actor lo solicita, el sistema muestra las métricas obtenidas del fichero solicitado.**Actores:** Usuario, Administrador**Precondiciones:**

- El fichero solicitado debe haber sido analizado previamente y las métricas obtenidas almacenadas en la base de datos.

Operaciones básicas	Actor	Respuesta del sistema
1.	Solicita el listado de los ficheros analizados (historial, notificaciones)	Muestra todos los ficheros subidos por el actor, añadiendo un enlace a aquellos que están analizados
2.	Pincha en un fichero de entre los disponibles listados previamente	Muestra las métricas del fichero solicitado
Alternativas		
1.A		Comprueba si el actor es Premium
1.A.1		Si sí, muestra todas las medidas y las gráficas resultantes
1.A.2		Si no, muestra únicamente las medidas básicas resultantes

Tabla 4. Flujo de eventos - Ver métricas de un fichero

Especificación del caso de uso**CU5:** Eliminar un fichero

Descripción: Elimina el documento indicado por el usuario, eliminando también todas las métricas asociadas en caso de haberlas.

Actores: Usuario, Administrador

Precondiciones:

- El fichero tiene que existir

Operaciones básicas	Actor	Respuesta del sistema
1.	Solicita el listado de los ficheros subidos (historial)	Muestra todos los ficheros subidos por el actor
2.	Selecciona la opción “eliminar” del fichero deseado, de entre los listados previamente	Elimina el fichero indicado y deja de mostrarlo en el listado

Tabla 5. Flujo de eventos - Eliminar un fichero

Especificación del caso de uso**CU6:** Modificar datos

Descripción: Permite al actor modificar los datos personales introducidos previamente en el CU1

Actores: Usuario, Administrador

Precondiciones: Ninguna.

Operaciones básicas	Actor	Respuesta del sistema
1.	Solicita acceder a los datos personales	Muestra los datos personales

2.	En el formulario mostrado, modifica aquellos campos que desee cambiar	
3.	Pincha en “actualizar perfil”	Almacena en la base de datos las modificaciones realizadas
Alternativas		
1.A		Comprueba si el actor es Administrador
1.A.1		Si sí, muestra un listado de todos los datos personales de todos los usuarios
1.A.2		Si no, muestra únicamente los datos personales del Usuario
3.A		Comprueba que todos los campos están completos.
3.A.1		Si sí, continúa
3.A.2		Si no, avisa al actor y vuelve a 2
3.B		Comprueba que las contraseñas introducidas sean iguales
3.B.1		Si sí, continúa
3.B.2		Si no, vuelve a 2, avisa al actor de que son diferentes
3.C		Comprueba que el username no existe

3.C.1		Si sí, vuelve a 2, avisa al actor de que ya existe, permite modificarlo
3.C.2		Si no, continuar. El actor ha modificado sus datos correctamente.

Tabla 6. Flujo de eventos - Modificar datos

Especificación del caso de uso**CU7:** Eliminar Usuario**Descripción:** Elimina al actor del sistema**Actores:** Usuario, Administrador**Precondiciones:** Ninguna

Operaciones básicas	Actor	Respuesta del sistema
1.	Solicita su eliminación del sistema	Elimina todos los ficheros asociados al actor y posteriormente, al actor.
2.		
Alternativas		
1.A		Comprueba si el actor es Administrador
1.A.1		Si sí, muestra un listado de todos los usuarios y le permite eliminar a cualquiera.
1.A.2		Si no, únicamente permite eliminar al propio actor

Tabla 7. Flujo de eventos - Eliminar usuario

Especificación del caso de uso

CU8: Modificar métricas**Descripción:** Modifica manualmente las métricas del fichero indicado.**Actores:** Administrador**Precondiciones:** Ninguna

Operaciones básicas	Actor	Respuesta del sistema
1.	Solicita al sistema un listado con todos los usuarios	Muestra todos los usuarios
2.	Elige el usuario deseado, aquel de cuyos documentos desea modificar las métricas	Muestra todos los datos del usuario seleccionado, los ficheros asociados y las métricas correspondientes
	En el formulario mostrado, modifica los valores de las métricas deseadas y pincha en “guardar”	Almacena las modificaciones correspondientes
Alternativas		
2.A		En caso de que las métricas no tengan valores, las muestra con valor vacío

Tabla 8. Flujo de eventos - Modificar métricas

5.1.2 Diagramas de secuencia

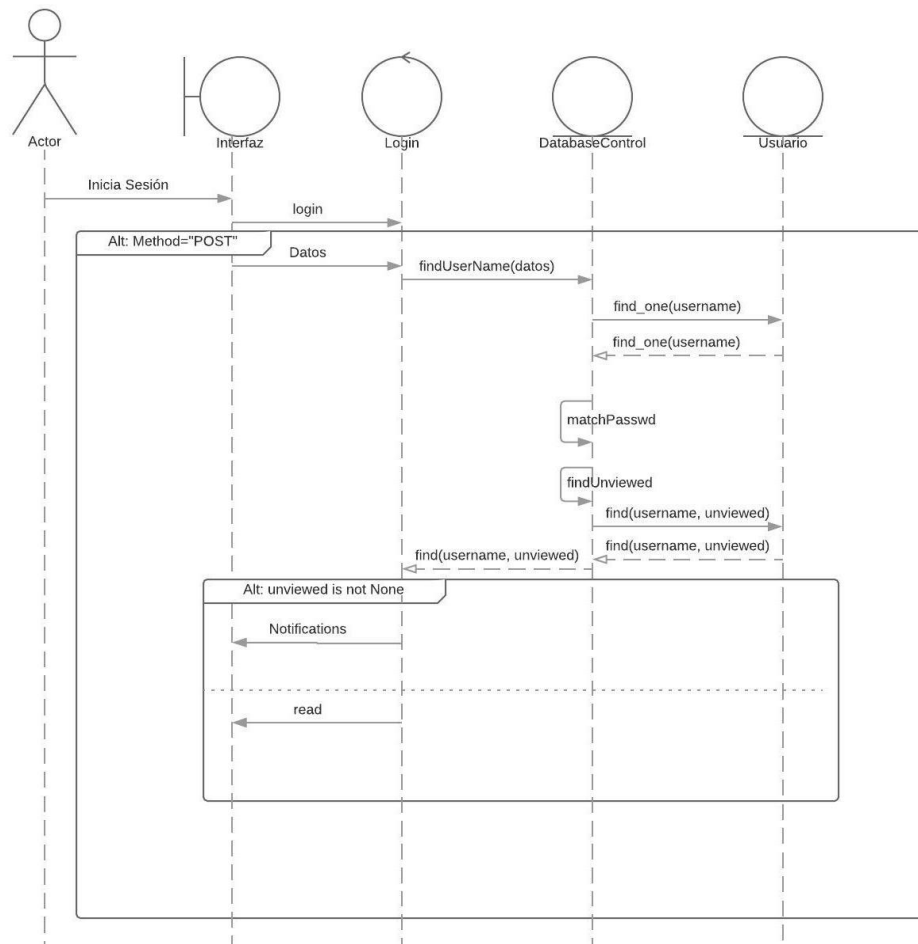


Ilustración 2. Diagrama de flujo - Iniciar sesión

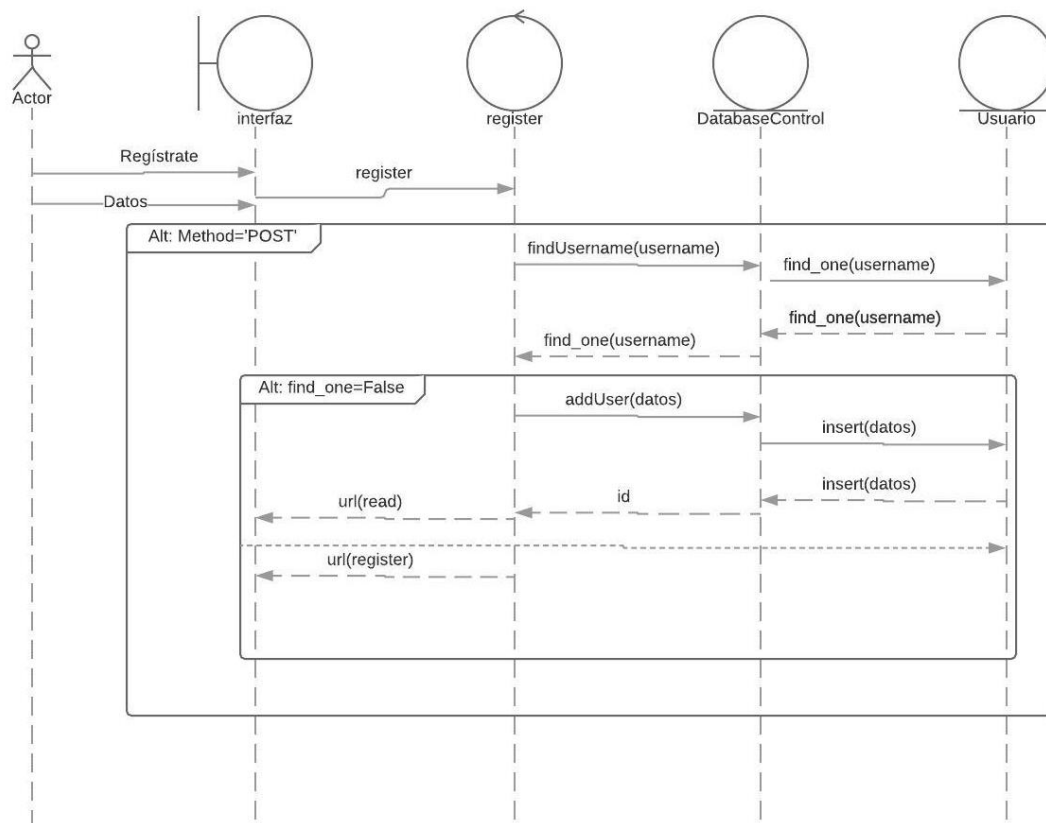


Ilustración 4. Diagrama de flujo - Registrar usuario

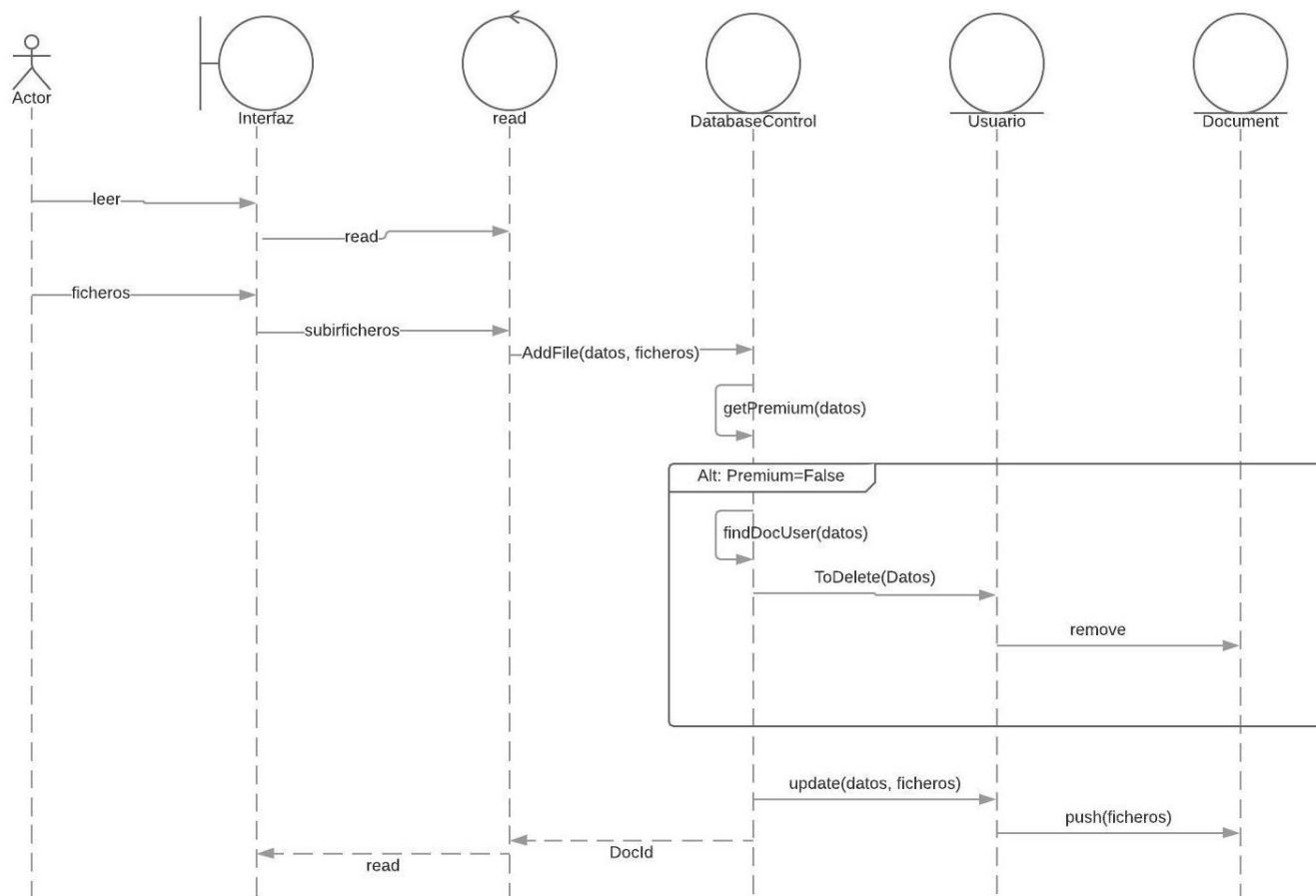


Ilustración 3. Diagrama de flujo - Subir fichero

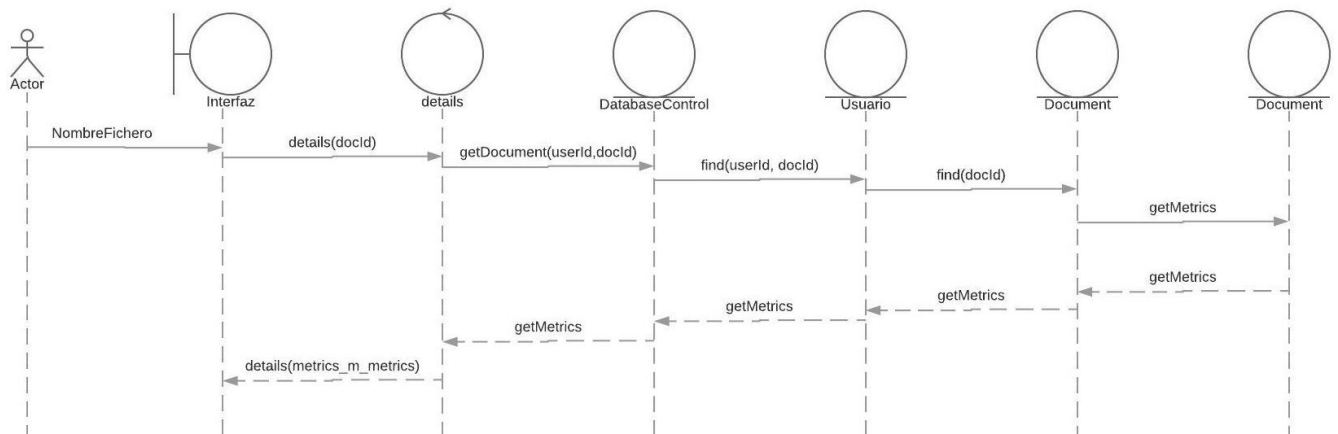


Ilustración 5. Diagrama de flujo - Ver métricas de un fichero

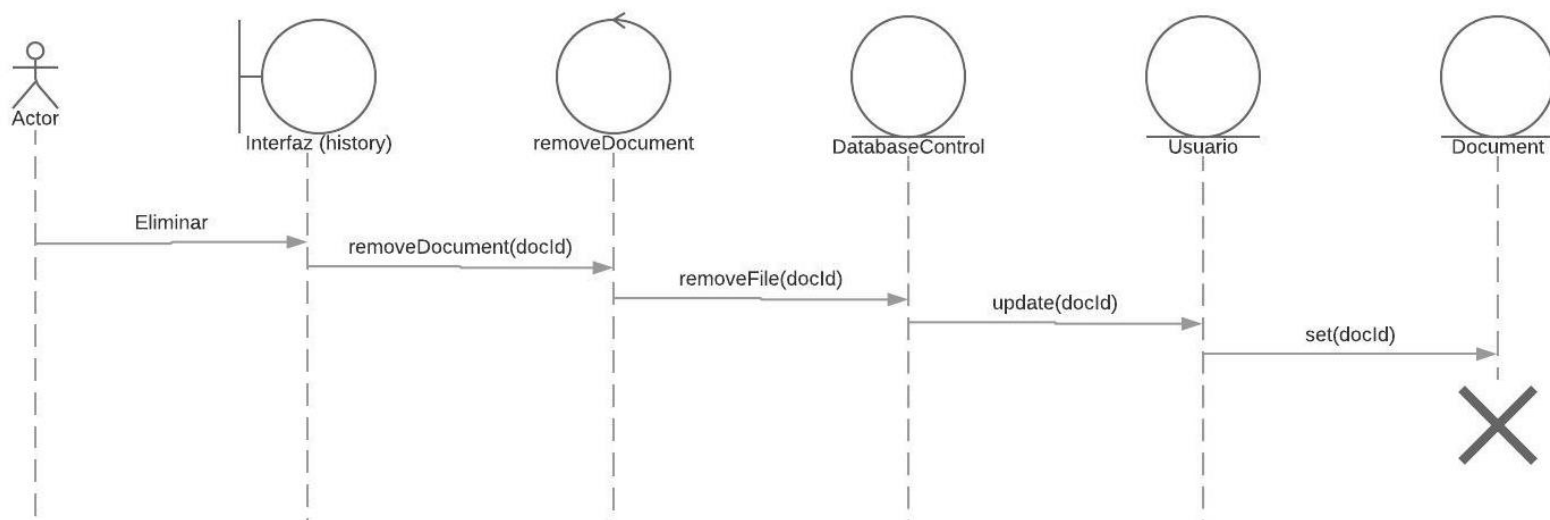


Ilustración 6. Diagrama de flujo - Eliminar fichero

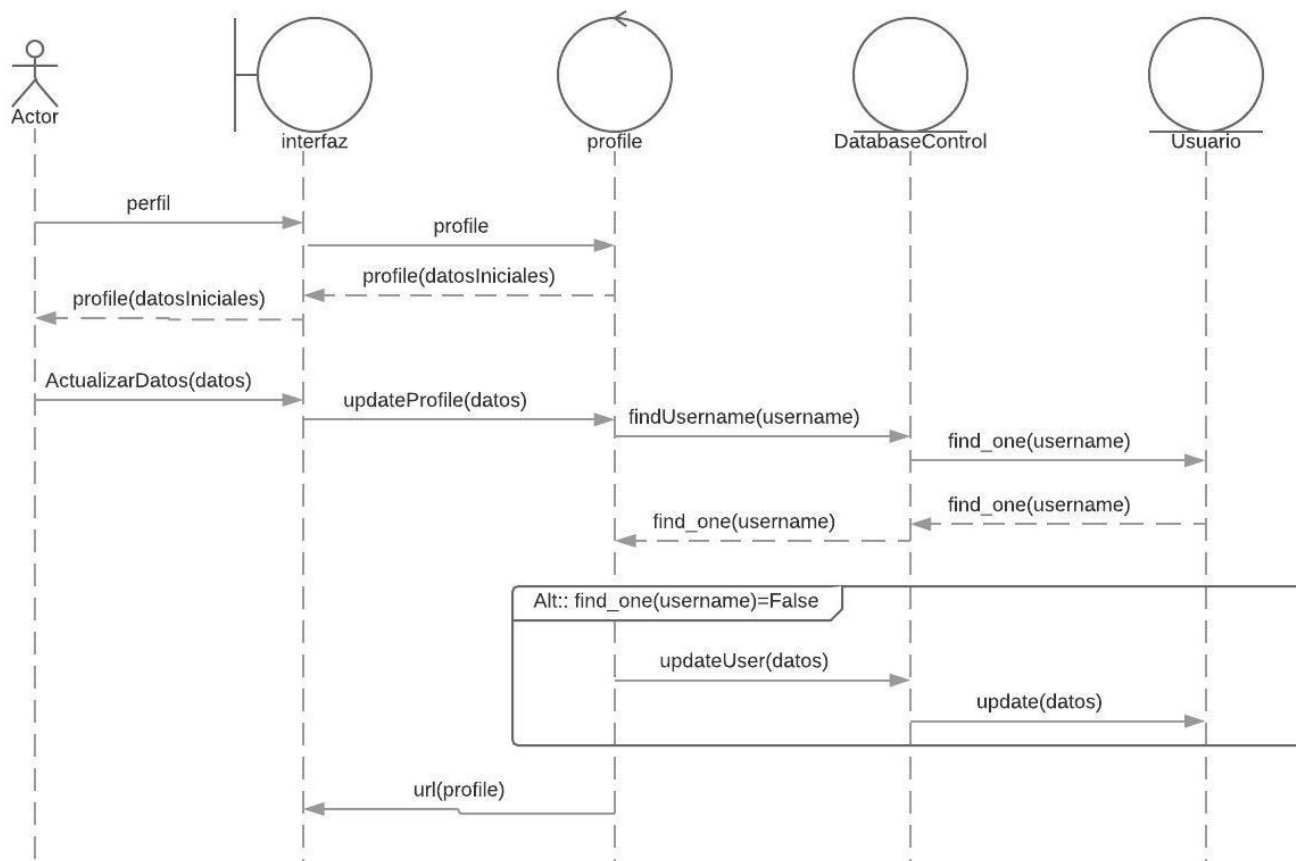


Ilustración 7. Diagrama de flujo - Modificar datos de usuario

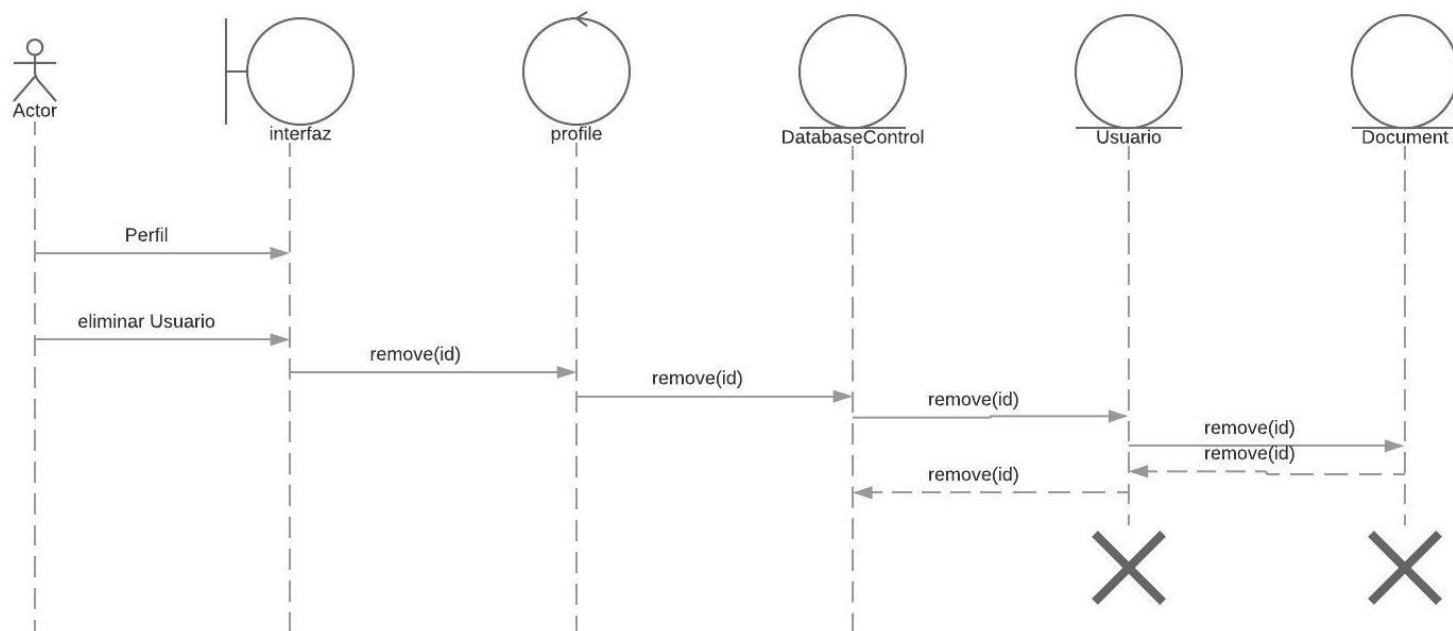


Ilustración 8. Diagrama de flujo - Eliminar usuario

5.2 Diseño

Para realizar un diseño inicial de la aplicación acorde a los distintos requisitos funcionales presentados, se ha procedido a la creación de modelos de simulación mediante mockups. Esta técnica consiste en la presentación de un producto gráfico no funcional, pero con la apariencia y el comportamiento de uno funcional, de forma que sea lo más fiel a la realidad posible. [2]

De esta forma el cliente se puede hacer una idea bastante completa de cómo será el producto final. El diseño de mockups nos permite también saber de forma más precisa cuáles son los requisitos por parte del cliente. Al utilizar esta técnica es muy importante dejar claro al cliente que con lo que está interaccionando no es el producto en sí, sino una simulación del mismo, y que puede sufrir modificaciones ligeras.

Se pretende que la aplicación sea lo más fiel posible a estos modelos de simulación iniciales, a menos que cambien los requisitos durante el desarrollo, por lo que la presentación final del producto puede variar respecto

Para la creación de mockups dinámicos se ha acudido a la herramienta “Balsamiq Mockups” en su versión de escritorio con licencia de estudiante, que permite la generación de una interfaz con distintas pantallas y transiciones entre ellas. [3]

5.2.1 Mockup

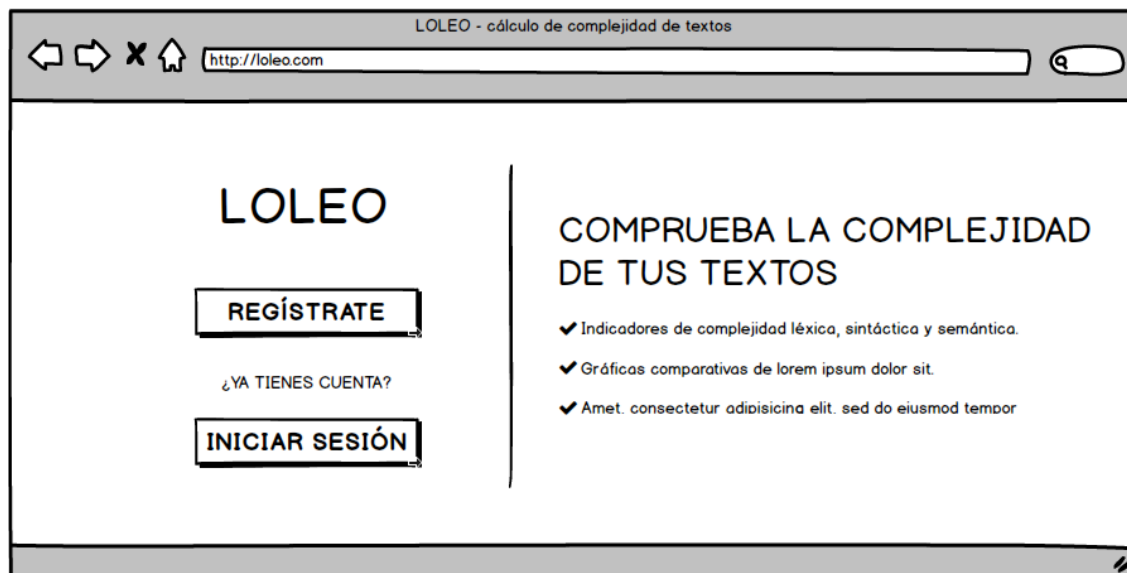


Ilustración 9. Mockup - Pantalla inicial

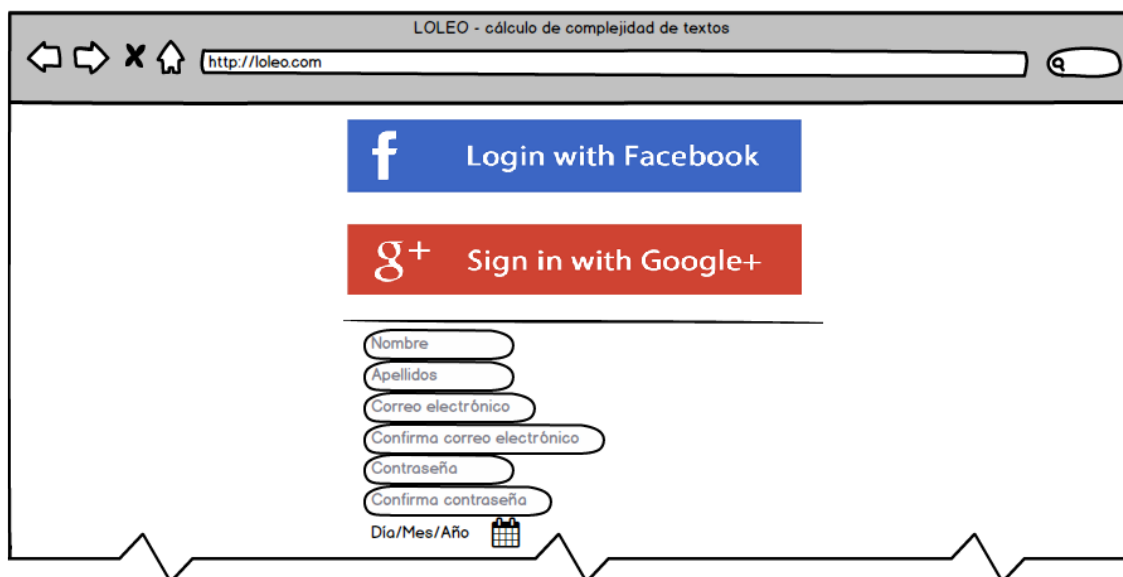


Ilustración 10. Mockup - Registro de usuario

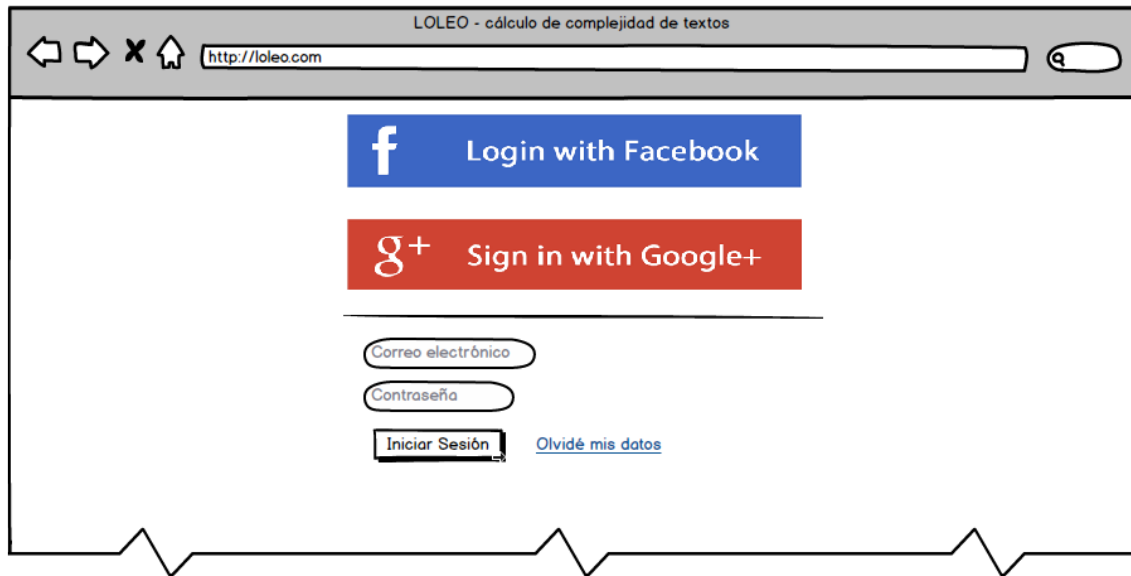


Ilustración 11. Mockup - Inicio de sesión

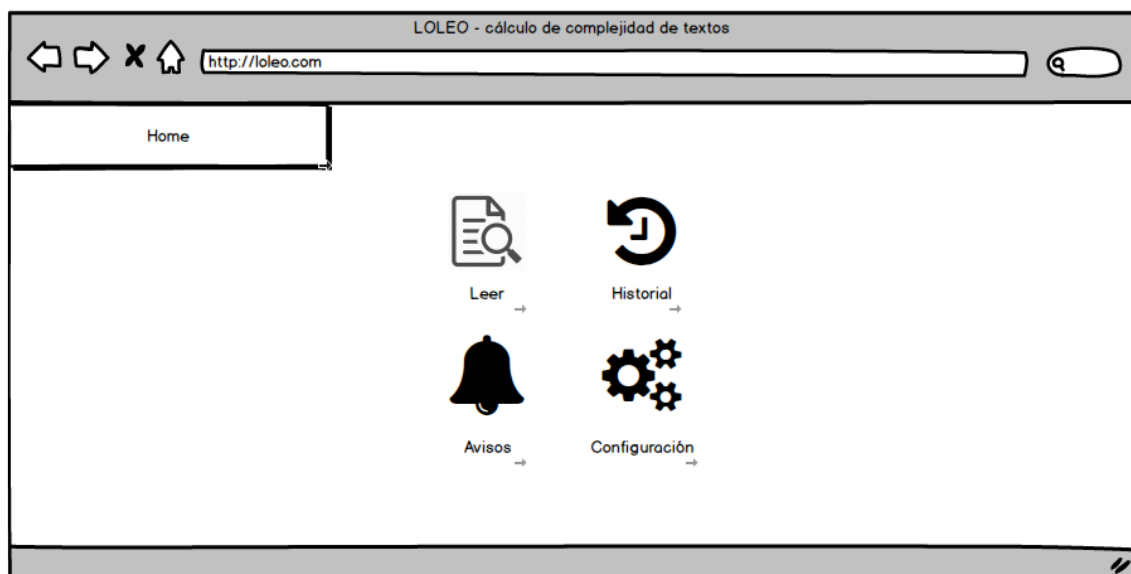


Ilustración 12. Mockup – MainMenu

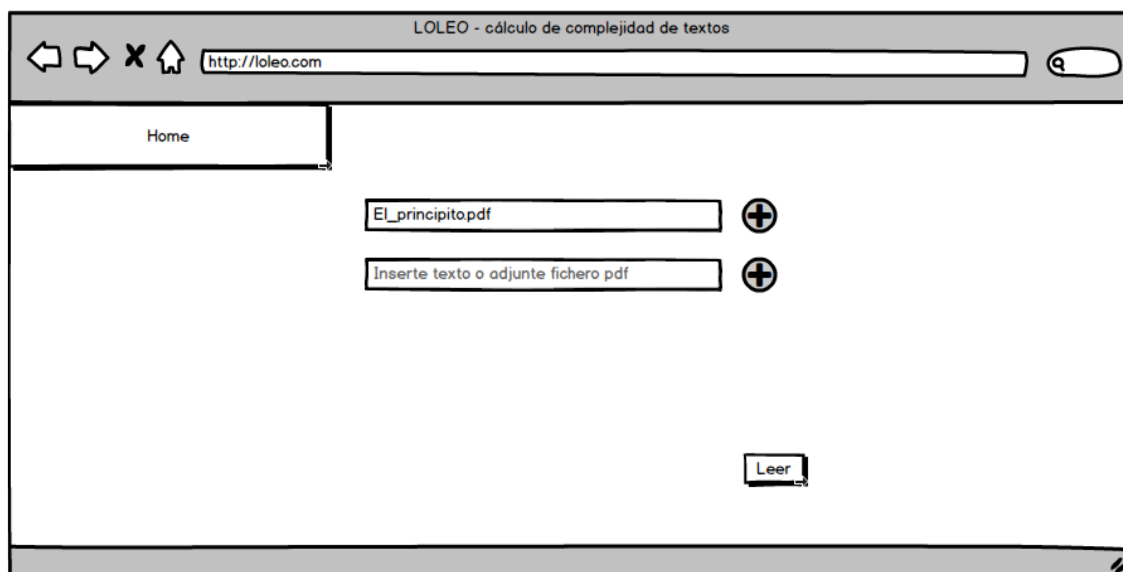


Ilustración 13. Mockup - Subir ficheros

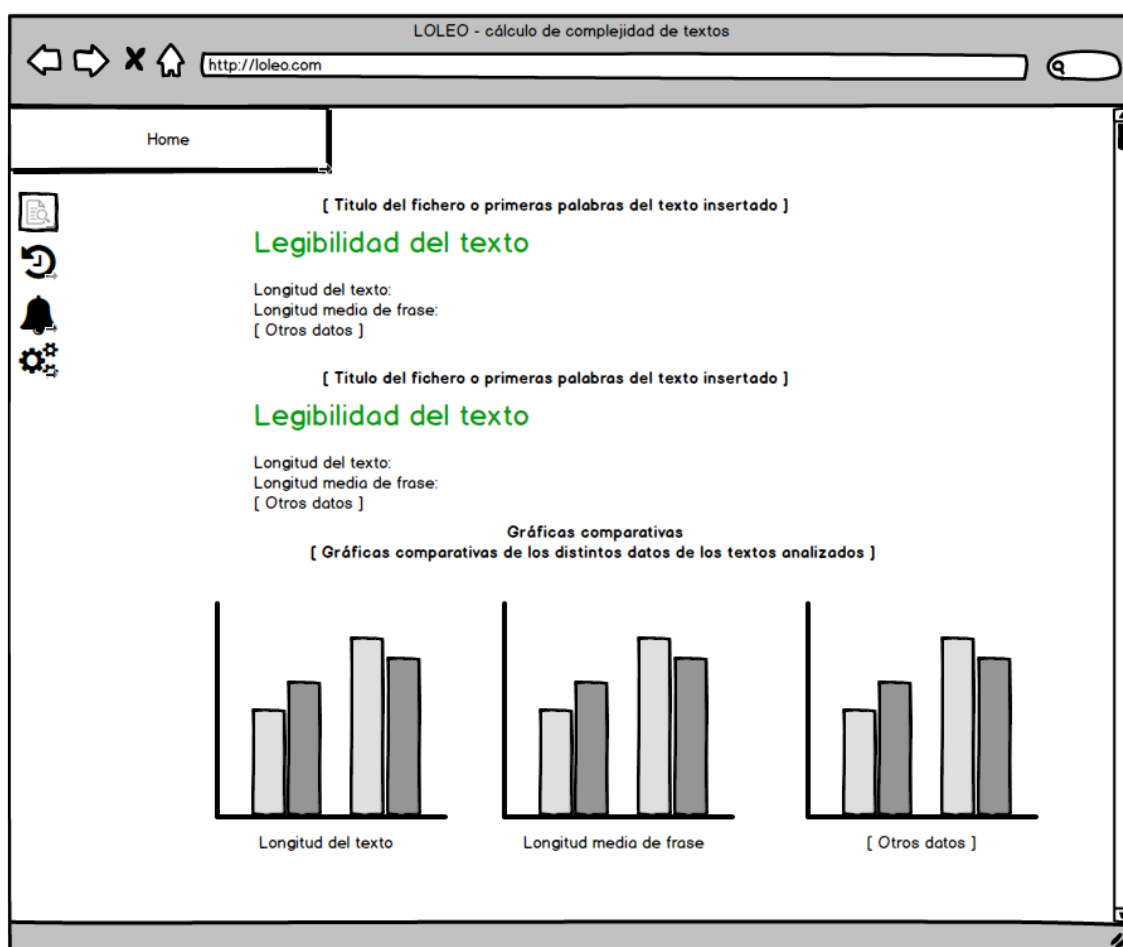


Ilustración 14. Mockup – Historial de documentos

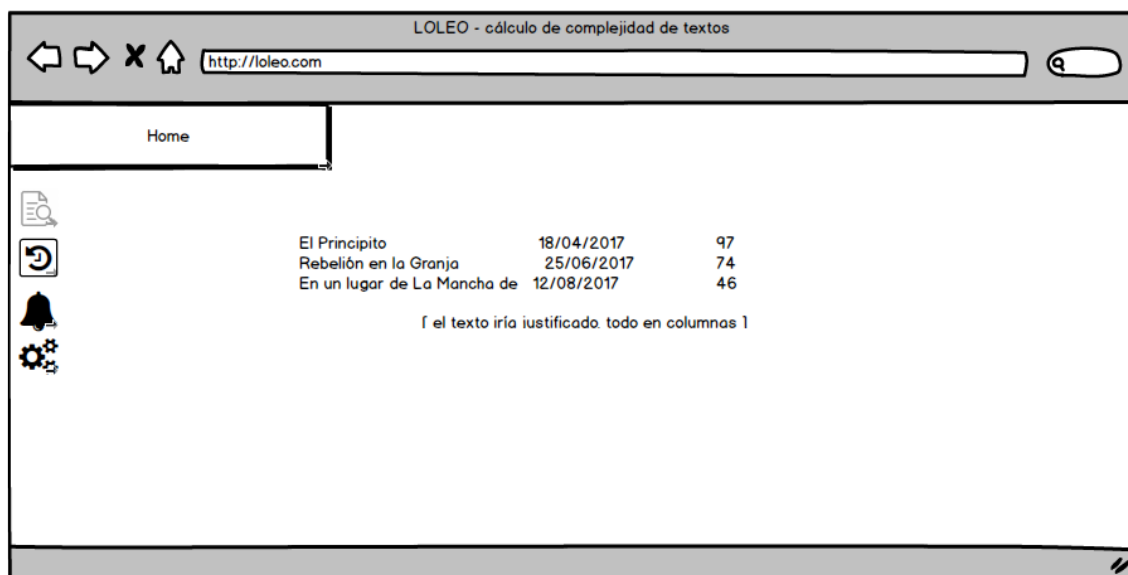


Ilustración 15. Mockup - Notificaciones

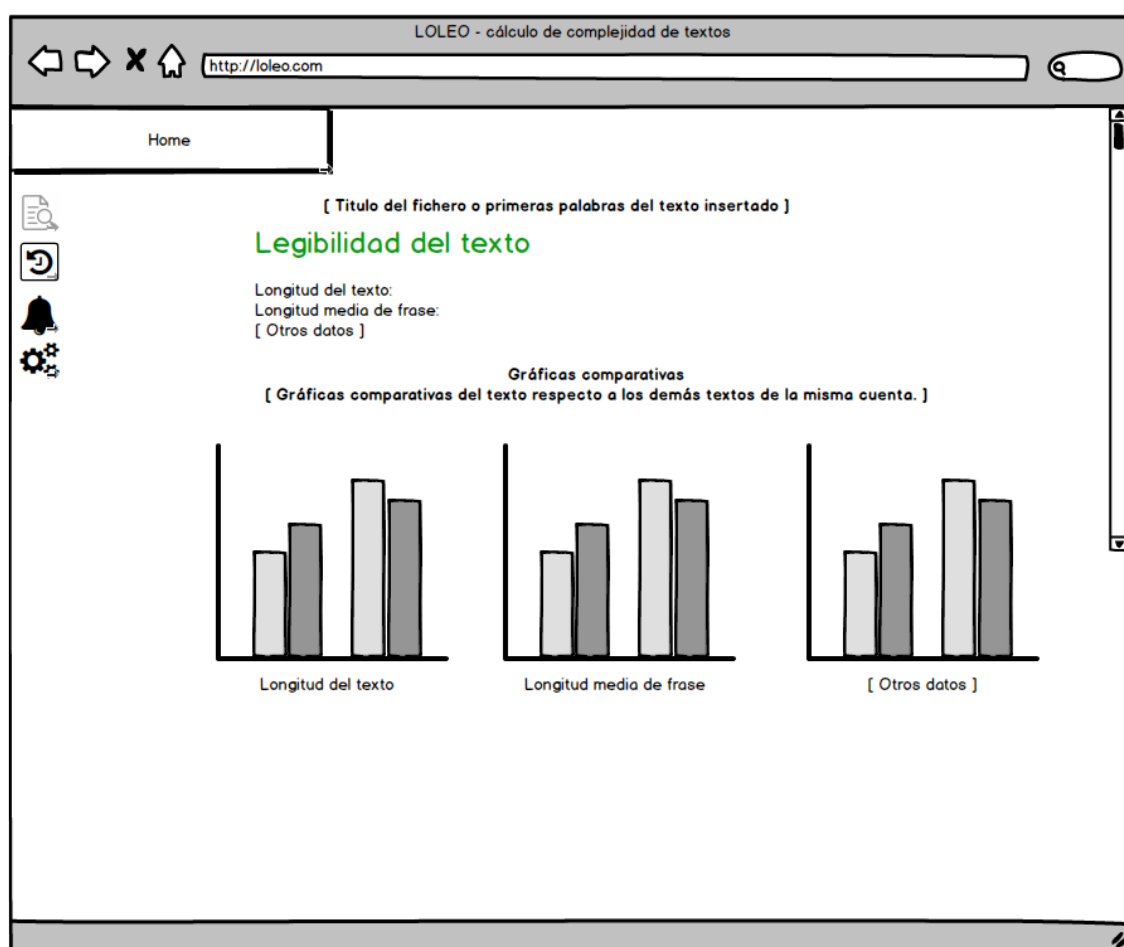


Ilustración 16. Mockup - Detalles del documento

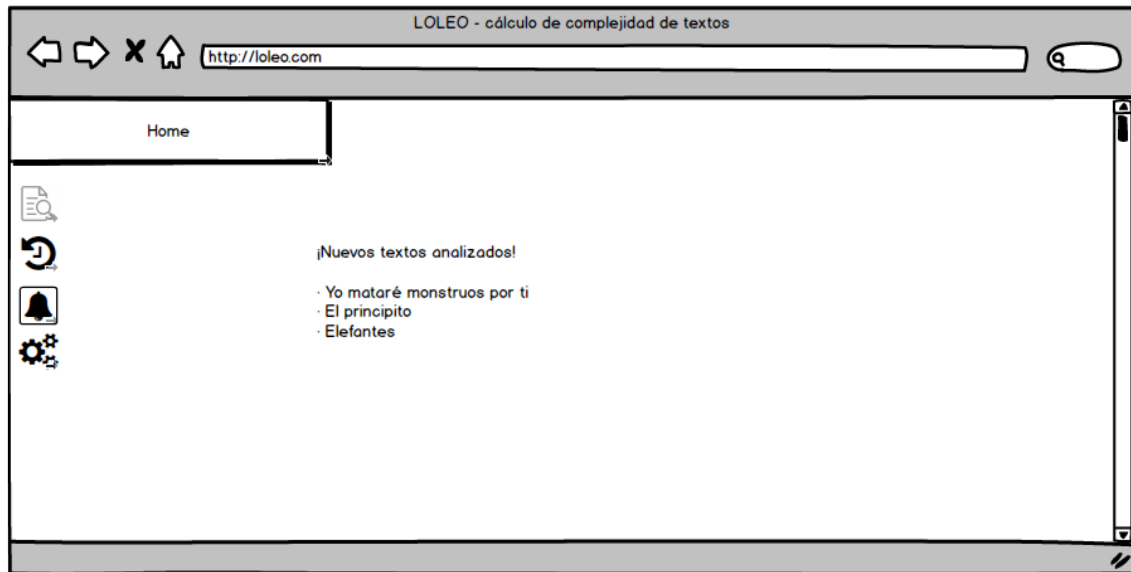


Ilustración 17. Mockup – Notificaciones

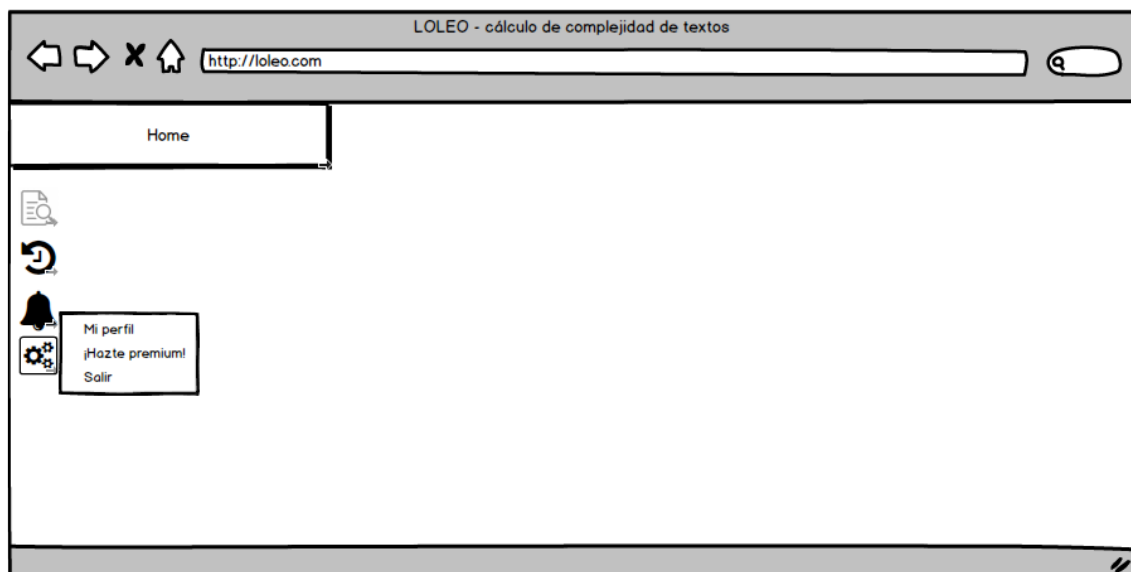
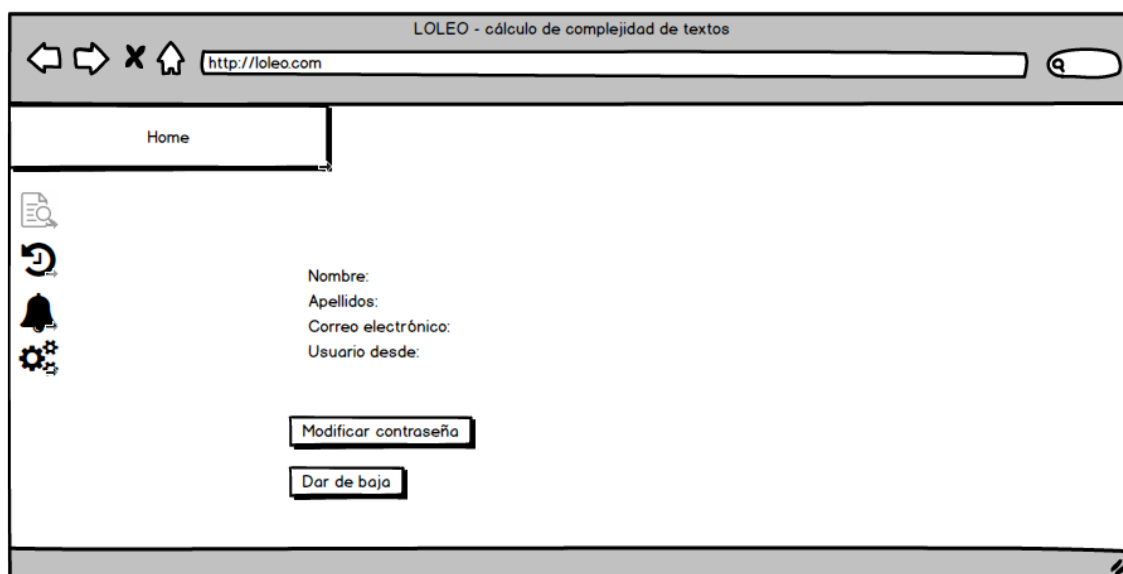
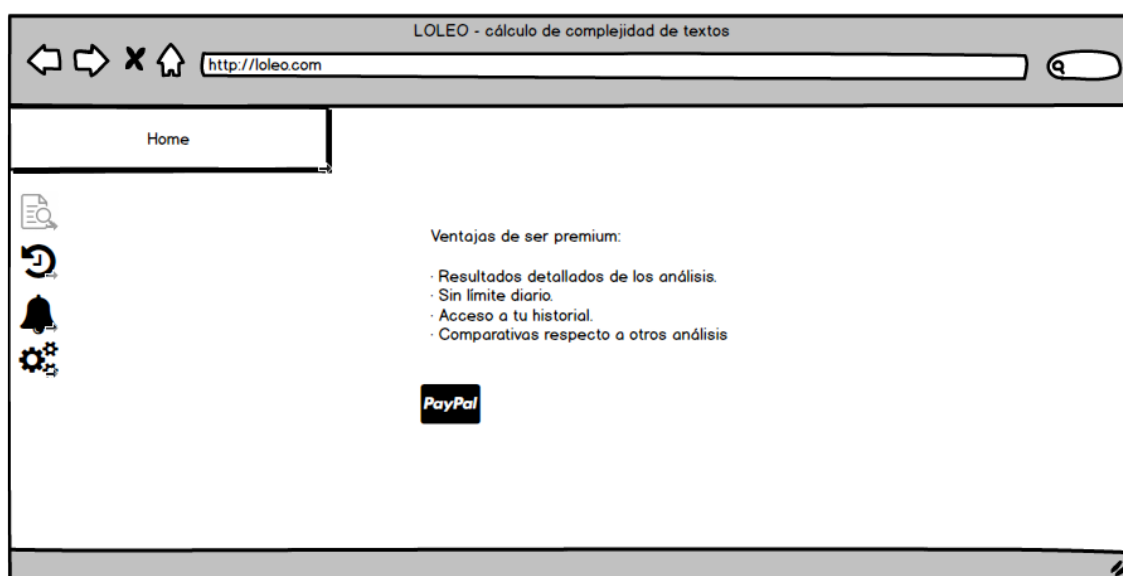


Ilustración 18. Mockup - Despliegue de configuración

*Ilustración 19. Mockup – Perfil**Ilustración 20. Mockup - Premium*

6. Tecnologías y lenguajes

El presente proyecto ha sido desarrollado principalmente sobre Python como lenguaje del servidor, el cual posee las librerías necesarias para poder desarrollar una aplicación con comunicación con clientes web.

Para el desarrollo de la interfaz del cliente se ha acudido al uso de HTML, lenguaje de marcado más utilizado y más soportado por los navegadores.

6.1 Python

Python es un lenguaje de programación de alto nivel, tiene un enfoque simple pero efectivo a la programación orientada a objetos. Cuenta con una sintaxis sencilla y muy dinámica. Por ello, es un lenguaje muy utilizado en desarrollo de aplicaciones en la mayoría de plataformas.

El intérprete de Python y la extensa biblioteca estándar están a libre disposición en forma binaria y de código fuente para las principales plataformas

El intérprete de Python puede extenderse con nuevas funcionalidades y tipos de datos implementados en C o C++.

Python ha sido utilizado en este proyecto para desarrollar todo el funcionamiento y comportamiento del sistema. A parte de los módulos que hay incluidos de base, otros han sido utilizados, los cuales están detallados a continuación. [4]

6.2 PyPdf2

Se trata de una librería de *Python* que permite la manipulación de ficheros en formato pdf.

En este proyecto en un principio estaba contemplada únicamente la posibilidad de trabajar con ficheros de texto sencillo (formato “.txt”), gracias a esta librería es posible la lectura y manipulación de ficheros en formato “.pdf”, ofreciendo así un mayor abanico de posibilidades al usuario en el momento de facilitar un documento para su análisis. [5]

6.3 MongoDB

MongoDb es un Sistema de bases de datos no relacionales basado en documentos. Las bases de datos no relacionales no almacenan los datos en forma de tuplas en tablas, sino, en este caso, como documentos agrupados en colecciones. En mongoDb los distintos atributos que pueda tener un objeto se denominan campos (field). Los datos son almacenados en lenguaje Bson.

Para su uso en Python se ha utilizado la distribución pyMongo, que provee las herramientas necesarias para la conexión y manipulación de la base de datos.

PyMongo se utiliza en la clase DatabaseControl(). Mediante esta clase se efectúan todas las conexiones e interacciones necesarias entre el sistema y la base de datos.[6]

6.4 XML

Para establecer los distintos parámetros de configuración del proyecto se ha utilizado un único fichero xml, lenguaje de marcado extensible. Mediante XML se pueden definir etiquetas personalizadas para descripción y organización de datos.

En el fichero desarrollado se han almacenado algunos parámetros personalizables, necesarios para el desarrollo de nuestra aplicación, tales como credenciales de conexión a la base de datos o directorios de almacenamiento de los ficheros manipulados. [7]

6.5 Freeling

La manipulación del texto para la obtención de las distintas medidas de complejidad léxica se ha llevado a cabo mediante esta herramienta a través de su API para Python.

Freeling es una librería desarrollada en c++ que suministra distintas funcionalidades para el análisis del lenguaje, tales como análisis morfológico o etiquetado sintáctico. [8]

Dadas las especificaciones del proyecto, se ha utilizado la versión en español de la herramienta.

La librería pyFreeling se utiliza para desarrollar la clase TextComplexityFreeling() [9]. Dicha clase utiliza el fichero “crea_total.txt”, donde se almacenan todos los términos y su frecuencia, para obtener algunos resultados de vocabulario relacionados con los términos menos frecuentes. [10]

6.6 Flask

Se trata de un microframework web con licencia BSD para Python basado en Werkzeug y Jinja2.

Al tratarse de un microframework, contiene únicamente lo necesario para que una aplicación web pueda funcionar, permitiendo extender su funcionalidad de la forma deseada. Hay una gran variedad de componentes disponibles para cualquier necesidad en Flask, tales como integración de bases de datos, manejo de subidas de ficheros o validación de formularios. [11]

Además, algunos módulos de esta librería se han utilizado para aumentar las funcionalidades del sistema:

- Flask_admin para desarrollar el backend de la aplicación web, al cual privilegios de acceso los administradores del sistema.

- Flask_paginate para mostrar al usuario los documentos recuperados de las distintas consultas de forma paginada

6.7 HTML

Como ya se ha adelantado anteriormente, HTML es el lenguaje de marcado más utilizado, soportado por la mayoría de navegadores. Su significado, *HyperText Markup Lenguaje*, determina que, para crear y representar visualmente una página web, utiliza un marcado mediante etiquetas. HTML define el contenido de la web pero no su funcionalidad, por lo que otras tecnologías distintas de HTML son utilizadas generalmente para describir la apariencia/presentación de una página web (CSS) o su funcionalidad (JavaScript). [12]

6.8 Bootstrap

Bootstrap es un framework front-end diseñado por *Twitter* para el desarrollo de aplicaciones y sitios web. Entre otras cosas, incluye CSS y HTML para tipografía, iconos, formas, distribución de elementos, capas y navegación. [13]

6.9 Fontawesome

Font awesome es un conjunto de herramientas de fuentes e iconos basado en CSS. Tiene su origen en *Twitter Bootstrap*. Dado que la versión 4 de *Bootstrap* no incluye ninguna librería de iconos por defecto, es necesaria su inclusión de forma independiente para poder mostrarlos correctamente. Es muy flexible ya que permite cambiar varios parámetros de estilo, tales como color, tamaño o fondo.

Dado que en su versión 4, bootstrap ya no incluía las mismas librerías de iconos que en versiones anteriores, ha sido necesario incluir esta librería de forma que se puedan utilizar los iconos necesarios. [14]

6.10 CSS

Se trata de un lenguaje desarrollado para describir el estilo de los documentos HTML, permitiendo personalizar su apariencia, describe cómo los elementos HTML deben ser mostrados en la interfaz.

CSS permite adaptar la presentación a distintos tipos de dispositivos, por ejemplo, pantallas grandes, pantallas pequeñas o diseño de impresión. CSS es independiente de HTML, y se puede utilizar con cualquier lenguaje de marcado basado en XML.

La separación de HTML y CSS hace que sea más fácil mantener sitios, compartir hojas de estilo entre páginas y adaptar páginas a diferentes entornos. [15]

6.11 JavaScript

Se trata de un lenguaje de alto nivel, desarrollado por *Netscape* y utilizado para la ejecución de scripts en páginas web principalmente, aunque también puede ser utilizado en otros proyectos que no requieran navegadores de internet, como *node.js* o *Apache CouchDB*. Posee una sintaxis muy similar a la de *Java* y *c++*, de ahí su definición.

JavaScript puede funcionar tanto como lenguaje procedural como lenguaje orientado a objetos. Los objetos se crean añadiendo métodos y propiedades a lo que de otra forma serían objetos vacíos en tiempo de ejecución. Una vez creado un objeto, es posible utilizarlo como prototipo para crear objetos similares.

La efectividad de *JavaScript* reside en las capacidades dinámicas que tiene, tales como creación de objetos en tiempo de ejecución, listas variables de parámetros o recuperación de código fuente. [16]

6.12 Google Charts

Es una librería de visualización de datos basada en *JavaScript* y desarrollada por *Google*, permite la creación y combinación de distintos tipos de gráficas manipulables creadas previa facilitación de datos por parte del usuario. Al estar basada en *JavaScript* puede ser ejecutado en cualquier navegador web que lo soporte [17]

7. Resultados

La realización del presente proyecto ha generado como producto final una aplicación web totalmente funcional conteniendo, además de los requisitos iniciales, las siguientes características:

- Obtención de tablas comparativas manipulables para cada usuario, mediante las cuales se pueden apreciar de forma resumida las distintas métricas y diferencias entre documentos.
- Rol Administrador, usuario con completo control tanto del sistema como de la base de datos.
- Implementación de un sistema de pago online mediante el cual se pueden obtener distintos roles del sistema.

8. Discusión

Durante el análisis y diseño de la aplicación web se desarrollaron algunas alternativas, tales como:

- Posibilidad de registro y login mediante las aplicaciones de Facebook y Google para usuarios que estén previamente registrados en dichas aplicaciones. Finalmente se ha descartado la posibilidad debido a que, al tratarse de una aplicación dirigida al ámbito laboral y educativo, nada tiene que ver con el ámbito social.
- Desarrollo de una pantalla principal para usuarios registrados. Se ha descartado también la opción, dado que de esta forma el usuario se ve obligado a emplear más clicks, empleando así más tiempo en realizar la tarea. En su lugar, se ha implementado el login de tal forma que la siguiente pantalla que aparezca es aquella que, con más seguridad, el usuario necesitará. De esta forma, muestra la pantalla de notificaciones para aquellos usuarios con documentos nuevos analizados, o muestra la pantalla de subida de ficheros en caso contrario.

9. Trabajo futuro

En la fase de análisis y diseño se recogieron algunas características que no han podido ser implementadas por falta de tiempo, las cuales quedan aquí plasmadas:

- Mensajes flash como respuesta del servidor tras la realización de algunas acciones, como eliminación de ficheros o modificación de datos personales.
- En el back-end, listado de todos los documentos en la base de datos, con posibilidad de ordenar por estado o fecha.
- En el back-end, añadir posibilidad de modificar contraseña de usuario, sin leer.

- Añadir funcionalidad de enviar correo electrónico con avisos de ficheros nuevos analizados.
- Añadir funcionalidad de envío de correo electrónico para restablecimiento de contraseña.
- Aunque la aplicación contiene tratamiento de errores, se puede ampliar implementando un sistema de registro de eventos y manejo de errores por parte del servidor.
- Desarrollo de la API correspondiente, de forma que se pueda interaccionar con la aplicación sin necesidad de uso de un navegador web.
- Dado que la aplicación posee un sistema de registro y tratamiento de datos personales, es conveniente la obtención de un certificado SSL.
- En la pestaña de detalles de un documento, dar la opción de ir directamente al documento anterior/siguiente en el historial

10. Bibliografía

- [1] Chowdhury, G. (2003) Natural language processing. Annual Review of Information Science and Technology
- [2] Blockwitz, T. (2011) The Functional Mockup Interface for Tool independent Exchange of Simulation Models
- [3] Balsamiq mockups
- [4] <https://www.python.org>
- [5] <https://pythonhosted.org/PyPDF2>
- [6] <https://docs.mongodb.com>
- [7] <https://www.w3.org/XML>
- [8] <http://nlp.lsi.upc.edu/freeling/index.php/node/16>
- [9] Montejo, A. – López, R. (2018) Complexity metrics as features for language modelling
- [10] <http://corpus.rae.es/lfrecuencias.html>
- [11] <https://www.w3.org/html/>
- [12] <https://flask.pocoo.org/>
- [13] <https://getbootstrap.com/docs/4.0>
- [14] <https://fontawesome.com/>
- [15] <https://www.w3c.es/Divulgacion/GuiasBreves/HojasEstilo>
- [16] <https://www.javascript.com/>
- [17] <https://developers.google.com/chart/>

11. Anexo: Manual de despliegue

1. Preparación del entorno	38
1.1. Instalación de Python y pip	38
1.2. Creación de un entorno con virtualenv	38
1.3. Instalación de MongoDB	39
1.4. Instalación de dependencias Python.....	40
1.4.1 Instalación de Freeling.....	40
2. Configuración de la aplicación	41
2.1. Fichero config.xml	41
3. Lanzamiento de la aplicación	42
4. Administración	42

1. Preparación del entorno

Para poder lanzar la aplicación, se debe preparar el entorno y dependencias necesarias

1.1. Instalación de Python y pip

Accede a la web de python.org, descarga la última versión ejecutable disponible, actualmente 3.6.4 y efectúa su instalación

1.2. Creación de un entorno con virtualenv

Abre un terminal de comandos y ejecuta la siguiente instrucción:

```
pip install virtualenv
```

Con esto, ya tenemos virtualenv instalado en nuestro equipo, por lo que podemos crear entornos virtuales donde instalar las dependencias necesarias para el proyecto sin afectar a la configuración del equipo. Teniendo Python y virtualenv instalados, nos dirigimos a la carpeta de nuestro proyecto desde un terminal de comandos, en la que ejecutaremos las siguientes instrucciones:

```
virtualenv flask
```

```
flask\scripts\activate.bat
```

1.3. Instalación de MongoDB

Descarga el instalador de MongoDB desde el centro de descargas de [mongodb.com](https://www.mongodb.com). Cualquier edición disponible es compatible con el proyecto. Versión utilizada 3.6.2. A continuación, ejecuta el instalador con derechos de administrador.

Crea las carpetas de almacenamiento y configuración de MongoDB

```
C:\data\db
```

Y dar permisos de lectura y escritura a estas carpetas.

A continuación, ya se puede iniciar el servicio del servidor de MongoDB mediante

```
C:\mongodb\bin\mongod.exe
```

También es posible ejecutar el servicio como un proceso mediante

```
Net start mongod
```

```
Net stop mongod
```

1.4. Instalación de dependencias Python

Una vez preparado el entorno, es necesario instalar las dependencias de librerías de Python, para ello, sobre el fichero de nuestro proyecto, ejecuta

```
pip install -r requirements.txt
```

1.4.1 Instalación de Freeling

Freeling es una herramienta utilizada por nuestra clase TextComplexityFreeling mediante su API para Python.

Para poder utilizarla, es necesaria su descarga desde <http://nlp.lsi.upc.edu/freeling/index.php/node/1> en la sección de descargas, o directamente en <https://github.com/TALP-UPC/FreeLing>. Para su correcta instalación, siga los pasos indicados en “INSTALL.windows”, teniendo en cuenta que al compilarlo es necesario habilitar el flag que genera la API para Python 3.

* Si durante la ejecución de Freeling se obtienen errores referentes al path, es necesario modificar el fichero “Analyze.bat”, estableciendo las primeras líneas del mismo como sigue:

```
set "BASEDIR=%~dp0"
```

```
set "OLDPATH=%PATH%"
```

```
set
```

```
PATH=%BASEDIR%\\..\\..\\dependencies\\boost\\lib;%BASEDIR%\\..\\..\\dependencies\\zlib\\bin;%BASEDIR%\\..\\..\\dependencies\\icu\\bin64;%BASEDIR%\\..\\..\\bin;%PATH%
```

```
set "FREELINGSHARE=%BASEDIR%\\..\\share\\freeling"
```

```
set cmdline=
```

Con esto, ya tendríamos todo lo necesario para que el proyecto pueda ejecutarse correctamente en el servidor.

2. Configuración de la aplicación

Aunque la aplicación puede ejecutarse directamente, es conveniente modificar algunos parámetros de configuración

2.1. Fichero config.xml

En este fichero se encuentran, separados en etiquetas, los distintos valores configurables de la aplicación.

- User: Nombre de usuario para la base de datos, en caso de que esté ubicada en un servidor diferente conviene establecer una autenticación.
- Password: Contraseña del usuario de la base de datos
- databaseHost: Host donde está ubicada la base de datos
- databaseName: Nombre de la base de datos dentro del sistema de base de datos utilizado
- databaseCollection: Colección utilizada en la base de datos para la aplicación. Únicamente se necesita una colección.

- uploadsFolder: Dentro del directorio de la aplicación, ruta en la que se almacenarán los ficheros enviados por los usuarios y manipulados para su análisis.
- Currency: Para el pago en Paypal, moneda de pago.
- amtToPay: Para el pago en Paypal, valor a pagar para conseguir la membresía premium
- secret: Utilizada para firmas de cookies y demás, conviene establecerla en una cadena aleatoria y compleja.

3. Lanzamiento de la aplicación

En primer lugar, asegúrese de que el servidor de MongoDB está en ejecución.

En distintas consolas, y dentro del directorio de la aplicación, ejecute los siguientes comandos para iniciar respectivamente el proceso de lectura de ficheros pdf, el proceso de análisis de documentos para la obtención de métricas y el proceso principal de la aplicación web.

```
py ReadPdfProcess.py
```

```
py AnalyzeProcess.py
```

```
py main.py
```

4. Administración

Una vez está la aplicación en funcionamiento, es necesario registrarnos en ella para obtener el usuario administrador.

El usuario Administrador será el primero en registrarse en la base de datos.

Para acceder al back-end de la aplicación es necesario acceder mediante [NombreDelSitio]/**admin**. Encontraremos dos pestañas distintas:

- Una referente a la base de datos, donde se muestran todos los usuarios registrados, los ficheros insertados y las métricas asociadas a los mismos. Aquí podremos modificar los datos almacenados en la base de datos, eliminar o añadir nuevos. Los valores de las contraseñas almacenadas no son accesibles.
- Otra en la que encontraremos un explorador de la carpeta de ficheros subidos al servidor, los cuales se podrán modificar.