



Escuela
Politécnica
Superior

Título del Trabajo Fin de Grado/Master



Grado en Ingeniería Multimedia

Trabajo Fin de Grado

Autor:

Nombre y Apellidos

Tutor/es:

Nombre y Apellidos

Marzo 2014



Universitat d'Alacant
Universidad de Alicante

Contrast

Subtítulo del proyecto

Autor

Adrián Tomás Vañó

Directores

Gustavo Candela Romero

Departamento de Lenguajes y Sistemas Informáticos

María Dolores Sáez Fernández

Departamento de Lenguajes y Sistemas Informáticos



GRADO EN INGENIERÍA MULTIMEDIA



Escuela
Politécnica
Superior



Universitat d'Alacant
Universidad de Alicante

ALICANTE, 4 de diciembre de 2018

Resumen

Se debe incluir en la memoria un resumen de máximo 500 palabras, mejor se hace al final del trabajo con la visión global del mismo. Es obligatorio presentarlo para el tribunal. Se entrega en un fichero a parte y también lo podemos incluir en la memoria. Puede estar en las tres lenguas cas val y en

Preámbulo

Poner aquí un texto breve que debe incluir entre otras:

“las razones que han llevado a la realización del estudio, el tema, la finalidad y el alcance y también los agradecimientos por las ayudas, por ejemplo apoyo económico (becas y subvenciones) y las consultas y discusiones con los tutores y colegas de trabajo. [AENOR, 1997]”

*A mi esposa Marganit, y a mis hijos Ella Rose y Daniel Adams,
sin los cuales habría podido acabar este libro dos años antes*¹

¹Dedicatoria de Joseph J. Roman en “An Introduction to Algebraic Topology”

*Si consigo ver más lejos
es porque he conseguido auparme
a hombros de gigantes*

Isaac Newton.

Índice general

Resumen	v
1. Introducción	1
2. Modelo de negocio	3
2.1. Lean Canvas	3
3. Objetivos	5
3.1. Generales	5
3.2. Específicos	5
4. Marco Teórico	7
4.1. Estado del mercado	7
4.1.1. Agregadores RSS	7
4.1.2. Servicios de seguimiento de prensa digital	7
4.2. Marco legal	8
4.2.1. Reglamento General de Protección de Datos	8
4.2.2. Explotación y uso de contenido de medios digitales	8
4.2.3. Conflicto AEDE con Google Noticias	8
4.3. Estudio de tecnologías para el desarrollo	8
4.3.1. Lenguajes de desarrollo	9
4.3.2. Back-end	10
4.3.3. Front-end	13
5. Metodología	15
6. Análisis y especificación	17
6.1. Especificación requerimiento funcionales	17
6.1.1. RF01: Obtener un listado de noticias recientes	17
6.1.2. RF02: Obtener un listado de noticias al realizar una búsqueda básica	17
6.1.3. RF03: Obtener un listado de noticias al realizar una búsqueda	
avanzada	17
6.1.4. RF04: Ampliar una noticia	17
6.2. Casos de uso	17
7. Diseño	19
7.1. Arquitectura seleccionada	19
7.2. Tecnologías	19

7.3. Diagrama de clases	19
7.4. Mockups	19
8. Implementación	21
8.1. Iteraciones?	21
9. Pruebas y validación	23
10. Resultados	25
11. Conclusiones	27
11.1. Conclusiones	27
11.2. Líneas de trabajo futuras	27
Bibliografía	29
A. Anexo I. Esquema de base de datos	31
B. Anexo II. Otros...	33

Índice de figuras

2.1. Lean Canvas de Contrast	3
4.1. Lenguajes más populares en la encuesta de 2018 de stackoverflow.com . .	9
4.2. Motores de búsqueda más populares en https://db-engines.com/en/ranking/search+engine	11
5.1. Espacio de trabajo para Contrast	15
5.2. Integración IFTT de Todoist con Trello	16

Índice de tablas

Índice de Listados

1. Introducción

En la introducción es donde se hará énfasis a la importancia de la temática, su vigencia y actualidad; se planteará el problema a investigar, así como el propósito o finalidad de la investigación, la aportación que supone este TFG al dominio/sector que se ha investigado. Debemos explicar el contexto donde se ubicará el trabajo.

No hay una longitud estimada, pero se debe tener en cuenta que es lo primero que lee una persona que debe evaluar el trabajo.

Se pueden establecer secciones si así lo requiere la memoria.

2. Modelo de negocio

2.1. Lean Canvas

Esta sección es opcional y depende de la tipología del trabajo a realizar. El siguiente apartado forma parte del estudio de viabilidad del proyecto, en él se debe reflejar y justificar el porqué surge el trabajo. Es importante pensar y plasmar en el trabajo cómo nuestro proyecto va a aportar valor a los clientes, conocer sus necesidades o problemas y cómo les damos solución y cómo estos clientes van a pagar por ese valor que reciben. Además, es importante conocer el tipo de mercado en el que operamos y qué otros factores influyen en él. Se analizan qué alternativas existen hoy en día en ese sector que solucionen total o parcialmente el problema y que afecten a los clientes o bien se ha detectado que no existe una solución (aunque sea comercial) a dicho problema.

Importante detectar problema o necesidad, a quién afecta, cómo se resuelve y qué se va a aportar, es decir cómo se va a solucionar (TFG).

Se pueden incluir análisis de DAFO y análisis de los riesgos

PROBLEM Dificultad para leer noticias de distintos periódicos en un mismo lugar Dificultad para contrastar información entre periódicos El seguimiento de medios para usos no-comerciales es caro EXISTING ALTERNATIVES Agregadores de medios que permiten la lectura de noticias Para contrastar la información hay que acceder a cada noticia por separado El seguimiento de medios se hace a través de empresas	SOLUTION Agregador de prensa digital de diferentes fuentes Visor simultáneo de noticias Seguimiento de medios básico, agrupando noticias según temática	UNIQUE VALUE PROPOSITION Visor para lectura simultánea de prensa digital	UNFAIR ADVANTAGE Lectura simultánea de noticias procedentes de distintos periódicos	CUSTOMER SEGMENTS Lectores de prensa digital
	KEY METRICS Búsquedas Accesos a la aplicación		CHANNELS Redes sociales Product Hunt	

Figura 2.1.: Lean Canvas de Contrast

3. Objetivos

3.1. Generales

El objetivo general del proyecto es crear un visor capaz de mostrar simultáneamente dos o más noticias que sean idénticas en temática, de tal manera que el usuario pueda contrastar la información.

3.2. Específicos

1. Construir una API que actúe como agregador de RSS.
2. Utilizar un motor de búsqueda para indexar el contenido RSS.
3. Utilizar una herramienta de análisis de lenguaje humano que determine la similitud entre noticias.
4. Diseñar un visor de noticias que permita la lectura simultánea.

4. Marco Teórico

La siguiente sección trata el estado de las tecnologías existentes para abordar la construcción de la aplicación, al mismo tiempo se mencionan las alternativas existentes que ofrecen un servicio similar al que plantea Contrast. Por otra parte, se trata el marco en el que se engloba el contexto de la aplicación.

4.1. Estado del mercado

En este apartado se detallan el conjunto de servicios sobre los que Contrast comparte nicho de mercado. Los servicios citados están disponibles en la web.

4.1.1. Agregadores RSS

Actualmente existen una gran cantidad de agregadores RSS con diferentes características y funcionalidades. Algunos muy populares entre los consumidores de estos servicios son Feedly, Flipboard e Inoreader. Dichos servicios permiten la búsqueda de fuentes de contenido para su lectura y la suscripción a temas de interés, que a su vez suscribe a varias fuentes relacionadas con el tema.

- **Feedly** es el lector de contenido más popular actualmente. Cuenta con una interfaz clara y simple y permite consumir contenido en diversos formatos: vídeo, noticias de periódicos, tweets, publicaciones de blogs, alertas Google Keywords y por supuesto, RSS. Posee herramientas de integración con servicios de terceros como Evernote o Trello y permite crear tableros compartidos con otros usuarios.
- **Flipboard** ha sido el agregador de noticias más utilizado durante mucho tiempo y una red social en sí. Posee integraciones con las principales redes sociales para leer contenido de ellas y compartir noticias en las redes desde la aplicación. Tiene un diseño cuidado que asemeja el uso de un magazín físico en cuanto a estructura de las secciones y el deslizamiento de hojas.
- **Inoreader** se define entre los usuarios como la alternativa a los anteriores cuando hay algún aspecto entre ellos que no termina de encajar con el usuario. Este lector de contenido posee herramientas para etiquetar y organizar automáticamente la información entrante en busca de ahorro de tiempo por parte del usuario.

4.1.2. Servicios de seguimiento de prensa digital

El seguimiento de medios consiste en la detección de toda información publicada sobre un tema concreto. Estos datos se analizan en pos de categorizarlos y mejorar la distri-

bución de los mismos. En el contexto en el que se encuentra Contrast, se estudian las empresas de seguimiento de prensa digital.

- **Pressclipping.com:** esta empresa ofrece análisis de medios y evaluación de noticias, además del propio seguimiento de medios. Cuenta además con soluciones para la gestión de las relaciones públicas y del marketing estratégico para medir con precisión el alcance y reputación de una marca.
- **Puntonews.com:** servicio que permite buscar noticias tanto de periódico, radio como de televisión y exportarlas a PDF.
- **MyNews.es:** hemeroteca digital desde 1996. Dispone de seguimiento de formatos audiovisuales y redes sociales, al igual que análisis de presencia y reputación de marca.

4.2. Marco legal

4.2.1. Reglamento General de Protección de Datos

4.2.2. Explotación y uso de contenido de medios digitales

4.2.3. Conflicto AEDE con Google Noticias

Google Noticias es un agregador y buscador de noticias de medios de prensa digital que cuenta con más de 70 ediciones en 35 idiomas y que genera más de 10.000 millones de clics al mes. Esta herramienta, nacida en 2002, no está disponible en España desde hace unos años.

El 5 de noviembre de 2014, el Partido Popular aprobó la nueva ley de propiedad intelectual (<https://www.boe.es/boe/dias/2014/11/05/pdfs/BOE-A-2014-11404.pdf>), causando que Google se viese obligado a pagar una tasa exigida por la Asociación de Editores de Diarios Españoles (AEDE). Dicho de otra manera, ahora tenía que pagar por mostrar cualquier fragmento del contenido de los editores españoles.

Cabe destacar que Google no tenía un modelo de negocio con este servicio, pues no mostraba anuncios, tan solo enlazaba los titulares con la noticia en el medio digital.

La ley aprobada entraba en vigor el 1 de enero de 2015 y el 16 de diciembre Google anunciaba que retiraba el servicio de noticias en España. Los editores contrastaron días después que la plataforma de Google les aportaba tráfico y por lo tanto ingresos por anuncios, por ello pidieron al gobierno y a la Unión Europea que intervinieran, abogando que el gigante tenía la posibilidad de negociar la tasa que debía pagar. Google fue tajante con su rechazo.

Actualmente la ley sigue en vigor.

4.3. Estudio de tecnologías para el desarrollo

En esta sección se detallan las tecnologías que puede necesitar el proyecto para su desarrollo. Está dividido en tres subsecciones. Por un lado se abarcan los lenguajes de

programación, por otro las herramientas pertenecientes al back-end y por último las que atañen al front-end.

4.3.1. Lenguajes de desarrollo

Se han analizado los lenguajes más populares en la última encuesta publicada en stackoverflow.com, que permiten tanto el desarrollo front-end como back-end, aunque se elijan opciones distintas para cada apartado.



Figura 4.1.: Lenguajes más populares en la encuesta de 2018 de stackoverflow.com

- Javascript fue el primer lenguaje de scripting para web. Es uno de los lenguajes más usados en web tanto para el back-end como para el front-end, integrado en la gran cantidad de frameworks que se han construido sobre él. Con la salida de ECMAScript 6 se han solucionado muchos problemas que acarreaba el uso de Javascript y se han añadido nuevas funcionalidades que lo convierten en aún más interesante. Tiene una de las bases de usuarios más sólidas entre todos los lenguajes.
- Java es un lenguaje con madurez y con muchos proyectos construidos sobre él que dependen de la instalación de Java en la máquina local. Se caracteriza por la solidez, confiabilidad, extensibilidad y rapidez. Existen multitud de frameworks y plugins que operan sobre Java y la documentación y comunidad es más que extensa. A este razonamiento se le suma el hecho de que es el lenguaje para el desarrollo en Android, por lo que es una opción muy interesante para aprender y trabajar en ella.

- Python es un lenguaje en alza desde los últimos años. Está diseñado partiendo de la intuitividad y del aproximamiento al lenguaje humano. La popularidad del lenguaje se debe al crecimiento del big data y del internet de las cosas, ya que ofrece algunas herramientas para el prototipado y las analíticas. Es un lenguaje ideal para principiantes, principalmente debido a su flexibilidad.

<https://www.quora.com/Which-language-has-the-best-future-prospects-Python-Java-or-JavaScript> <https://stackshare.io/stackups/java-vs-javascript-vs-python>

4.3.2. Back-end

El siguiente apartado contiene un sumario de tecnologías que posibilitan el back-end del proyecto. Por cada una de estas, se han detallado dos o tres propuestas en función de la popularidad, extensión y uso.

Parseador RSS

Dado que el proyecto debe obtener contenido RSS de distintas fuentes, necesita por lo tanto un transformador del contenido en un formato único con el que pueda trabajar el indexador. En este caso, se ha analizado la herramienta más popular para este cometido en cada uno de los lenguajes citados anteriormente.

- `rss-parser` es una librería mínima para convertir contenido RSS XML en objetos Javascript. Se compone de una función asíncrona que utiliza en un capa inferior un conversor de XML a objeto Javascript llamado `xml2js`. <https://www.npmjs.com/package/rss-parser>
- Universal Feed Parser es un módulo Python para descargar y parsear feeds. Es un plugin simple, autocontenido en un único fichero y con una única función primaria para parsear. Tiene una documentación extensa y funciones adicionales para sanitizar y parsear otros elementos contenidos en el feed. Trata también con el formato Atom. <https://pythonhosted.org/feedparser/index.html>
- Rome es un framework de Java de código abierto para tratar con RSS y Atom. Incluye parseadores, generadores y conversores entre ambos formatos. Los parseadores, el objeto de interés de Contrast, devuelve generalmente un objeto genérico que permite trabajar sin importar el formato de entrada o salida. Está compuesto por distintos módulos según se realice una acción concreta u otra. Posee una documentación sólida y es ampliamente utilizado para este fin. <https://rometools.github.io/rome/index.h>

Motores de búsqueda e indexadores

El contenido obtenido mediante los feeds anteriormente, debe ser almacenado e indexado, de tal manera que la búsqueda y recuperación de estos datos sea rápida y eficiente. En este sentido es de especial interés el uso de un sistema de recuperación de información con índice invertido, caracterizado por ser una estructura de datos capaz de manejar grandes

volúmenes de información orientados a texto. En esta categoría se han investigado los tres motores de búsqueda más populares. <http://www.sisorbe.com/ExamenFinalRI/invertidos.html>

18 systems in ranking, December 2018

Rank			DBMS	Database Model	Score		
Dec 2018	Nov 2018	Dec 2017			Dec 2018	Nov 2018	Dec 2017
1.	1.	1.	Elasticsearch	Search engine	144.70	+1.24	+24.92
2.	2.	3.	Splunk	Search engine	82.18	+1.81	+18.39
3.	3.	2.	Solr	Search engine	61.35	+0.47	-4.95
4.	4.	4.	MarkLogic	Multi-model	14.28	+0.81	+3.13
5.	5.	5.	Sphinx	Search engine	7.82	+0.46	+1.79
6.	6.	6.	Microsoft Azure Search	Search engine	5.67	+0.36	+1.56
7.	7.	7.	Algolia	Search engine	3.62	-0.13	+0.57
8.	8.	9.	Google Search Appliance	Search engine	2.93	+0.03	+0.20
9.	9.	8.	Amazon CloudSearch	Search engine	2.64	-0.11	-0.09
10.	10.	10.	Xapian	Search engine	0.60	-0.08	-0.02
11.	11.	11.	CrateDB	Multi-model	0.44	-0.06	-0.15
12.	12.	12.	SearchBlox	Search engine	0.24	0.00	+0.01
13.	13.	14.	searchxml	Multi-model	0.09	-0.01	+0.09
14.	14.	13.	DBSight	Search engine	0.06	+0.00	+0.05
15.	15.	14.	Manticore Search	Search engine	0.04	-0.01	+0.04
16.			FinchDB	Multi-model	0.03		
17.	16.	14.	Exorbyte	Search engine	0.01	+0.01	+0.01
18.	16.	14.	Indica	Search engine	0.00	±0.00	±0.00

Figura 4.2.: Motores de búsqueda más populares en <https://db-engines.com/en/ranking/search+engine>

Apache Solr y Elasticsearch están construidas sobre Apache Lucene. Lucene es una potente librería de recuperación de información. Se encarga de construir el índice invertido mencionado anteriormente, una estructura especializada en emparejar documentos de texto con términos de consulta. Lucene, al igual que otros motores de esta clase, se distingue por la escalabilidad, el rápido despliegue, el manejo de grandes volúmenes de datos, la optimización orientada a búsqueda y esta a su vez, en textos.

Solr es un servidor construido con la librería de Lucene, siendo ambos proyectos de Apache. Solr provee una interfaz web de administración, APIs en distintos lenguajes para la realización de consultas y una serie de mejoras funcionales sobre las características base de Lucene.

Elasticsearch también se nutre de Lucene. Esta herramienta ha basado su modelo en una API REST Web al ser un motor de búsqueda más reciente y por ello ha ganado popularidad desde la última mitad de década. No existen grandes diferencias entre esta herramienta y la anterior, pero algunas destacables son:

- Solr posee una documentación más extensa y de mayor calidad al ser open source y llevar muchos años disponible.
- Elasticsearch tiende a tener una mejor ejecución en las consultas analíticas y Solr en las orientadas a texto.

- Elasticsearch es más intuitivo que Solr, especialmente para desarrolladores que se inician en este tipo de tecnologías.
- Elasticsearch tiene una escalabilidad inherente más fácil de conseguir que Solr, que se desarrolla mediante SolrCloud.

Por otra parte Splunk es un motor de búsqueda y plataforma orientado a las analíticas para el Big Data, es por ello que se ha desestimado su análisis.

Herramientas de procesamiento de lenguajes naturales

Para determinar que un grupo de noticias son similares en distintos periódicos, no solo basta con realizar una búsqueda por tema, también es necesario ver si ese tema en concreto pertenece a un mismo hecho o a distintos. Es por ello que Contrast necesita incorporar una herramienta de procesamiento de lenguaje natural. Este tipo de herramientas realizan análisis morfológicos, sintácticos, semánticos y pragmáticos. En el caso de Contrast, los más interesantes son los análisis semántico y pragmático, dado que el interés recae en conocer las temáticas generales y la intención de las noticias. Existen muchas propuestas en este ámbito, entre ellas destacan: Stanford CoreNLP, Apache OpenNLP y Freeling.

- CoreNLP es un proyecto de la Universidad de Stanford que provee una serie de herramientas de análisis de lenguaje humano. Proporciona instrumentos para un amplio análisis gramático, además de extractores muy diversos. Entre estos se halla uno de especial interés para el proyecto, la herramienta de reconocimiento de entidades (NER). CoreNLP posee una documentación extensa ya que es ampliamente utilizado, complementándose además con actualizaciones frecuentes que mejoran el producto. Posee soporte oficial para seis idiomas (entre ellos el español) y una serie de APIs en distintos lenguajes para una integración sencilla en diferentes proyectos. Esta herramienta es utilizada por la Biblioteca Virtual Miguel de Cervantes, concretamente el analizador sintáctico <http://data.cervantesvirtual.com/analizador-sintactico-automatico>.
- OpenNLP es un proyecto Apache, consistente en una librería de aprendizaje automático cuyo objetivo es el procesamiento del lenguaje humano en texto. Posee soporte para las tareas NLP más comunes. No tiene soporte al uso de distintos lenguajes, pero posee una serie de modelos entrenados en diferentes idiomas para propósitos determinados. En el caso del español encontramos cuatro modelos entrenados en reconocimiento de entidades: personas, organizaciones, localizaciones y misceláneo. Cualquier extractor de interés que no esté en los modelos tendría que ser entrenado para su propósito.
- Freeling es un proyecto de la Universitat Politècnica de Catalunya. Se trata de una librería con una serie de herramientas open source para el análisis de lenguaje humano. Está escrita en C++ y posee una extensa variedad de utilidades, entre ellas las de interés para el proyecto. Tiene soporte para muchos idiomas y dialectos

españoles, pero carece de una comunidad sólida que utilice la herramienta de manera asidua, al igual que una documentación no tan extensa como las propuestas anteriores.

Frameworks de desarrollo

Para esta sección se ha detallado un framework por cada lenguaje que se ha analizado previamente. Esta selección de frameworks ha sido realizada bajo criterios de popularidad, mantenimiento, calidad en la documentación, extensibilidad e integración con el resto de herramientas que componen el entorno de Contrast.

- Spring Boot es un framework basado en Spring para Java. Abstrae el proceso de elección de dependencias y despliegue del proyecto, realizando en cambio un proceso sencillo y automatizado. El objetivo principal de esta herramienta es proporcionar un conjunto de herramientas para construir rápidamente aplicaciones en el potente entorno Spring. Se trata de un framework maduro, con resultados profesionales e integraciones con otros productos de Spring que añaden más funcionalidades. Cabe destacar que posee una capa de abstracción para el motor de búsqueda Solr. <https://spring.io/>
- Express.js es un framework rápido, minimalista y flexible para el entorno de ejecución Node.js para Javascript. Proporciona un conjunto sólido de características para realizar APIs a través de métodos y utilidades HTTP y middleware que garantizan una creación rápida y sencilla. <https://expressjs.com/>
- Django es un framework full-stack para Python que facilita rapidez y claridad, ya que se hace cargo de muchos conceptos que conlleva el desarrollo web. Es una herramienta altamente escalable debido a su flexibilidad y proporciona ayuda a los desarrolladores para evitar muchos errores comunes de seguridad. <https://www.djangoproject.com/>

4.3.3. Front-end

El siguiente apartado contiene un resumen de tecnologías que posibilitan el front-end y diseño de interfaces del proyecto. Por cada una de estas, se han detallado dos o tres propuestas en función de la popularidad, extensión y uso.

Frameworks de desarrollo

Dentro de los frameworks para el desarrollo del front-end, se puede escoger una herramienta de un lenguaje distinto al del back-end. En este caso, se detalla un conjunto de frameworks con gran popularidad en la actualidad. Todos ellos basados en Javascript, el lenguaje de front-end que más alcance tiene estos días.

- Angular es una plataforma para diseñar aplicaciones multidispositivo creada por Google que facilita el desarrollo de aplicaciones web SPA, proporcionando herramientas para trabajar con los elementos de la página web de una manera óptima.

Tiene una gran comunidad detrás y una documentación extensa, pero en cierto modo fuerza al usuario a aprender TypeScript. <https://angular.io/>

- Vue es un framework progresivo con un ecosistema que se plantea cercano, versátil y con gran rendimiento. Esta herramienta, diseñada para construir interfaces de usuario reactivas, se creó con el propósito de ser fácil de integrar con otras librerías o proyectos a través del desacople de las partes que lo conforman. Es un framework relativamente nuevo que mejora algunos conceptos de Angular y React, a través de una documentación clara y detallada. <https://vuejs.org/>
- React es una librería para construir interfaces de usuario reactivas SPA. Esta herramienta de Facebook permite la creación de aplicaciones web potentes que requieran un intercambio de datos constante. Tiene una curva de aprendizaje algo elevada que recompensa con eficiencia y responsividad. <https://reactjs.org/>

Librerías de diseño

En este apartado se detallan tres librerías de diseño de interfaces, que complementan el front-end con animaciones y elementos preconstruidos. La selección ha sido realizada en base al número de estrellas que acumula cada proyecto en Github.

- Bootstrap es la librería de diseño más popular para construir aplicaciones web responsive pensadas en el mobile first. Dispone elementos que al utilizarlos aplican diferentes propiedades HTML, CSS y JS con un esfuerzo mínimo. Posee una gran comunidad detrás que además contribuye con plugins extra.
- Foundation es una librería open-source que incluye semántica. Su uso es intuitivo, complementándose con entrenamiento, soporte y consultas, por lo tanto garantiza una gran ayuda para los desarrolladores. Tiene además la capacidad de ser utilizado para diseñar plantillas de correo electrónico.
- Semantic UI posee el enfoque en realizar más semántico el proceso de construcción de una página. La principal funcionalidad consiste en utilizar principios del lenguaje natural para hacer el código más legible y fácil de entender. Posee una documentación bien organizada y con múltiples guías para empezar.

5. Metodología

Este proyecto, con un presupuesto de 300 horas ha sido estimado para ser realizado en dos meses y medio. De esta manera quedan garantizadas seis horas diarias durante cinco días a la semana, teniendo los fines de semana para recuperar alguna hora que por algún motivo, no se haya podido realizar entre semana.

Se ha seguido un modelo de desarrollo incremental mediante iteraciones cada 10-15 días. Al finalizar cada iteración se ha hecho una revisión con los tutores del proyecto con el fin de analizar el avance, responder dudas surgidas durante el sprint y planificar el siguiente. El trabajo realizado cada iteración se detalla en el apartado 8 de la página 21.

A continuación se detallan las herramientas que se han empleado para la organización del proyecto:

- Workona: extensión del navegador Chrome que facilita la gestión de las pestañas. Su principal funcionalidad es crear espacios de trabajo en el que se guardan las pestañas actualmente abiertas de una sesión a otra. Esto permite cambiar el dispositivo de trabajo fácilmente sin perder el contenido que se estaba visitando en ese momento. Al mismo tiempo también permite cambiar el proyecto en el que se esté trabajando con un solo clic. <https://workona.com/>

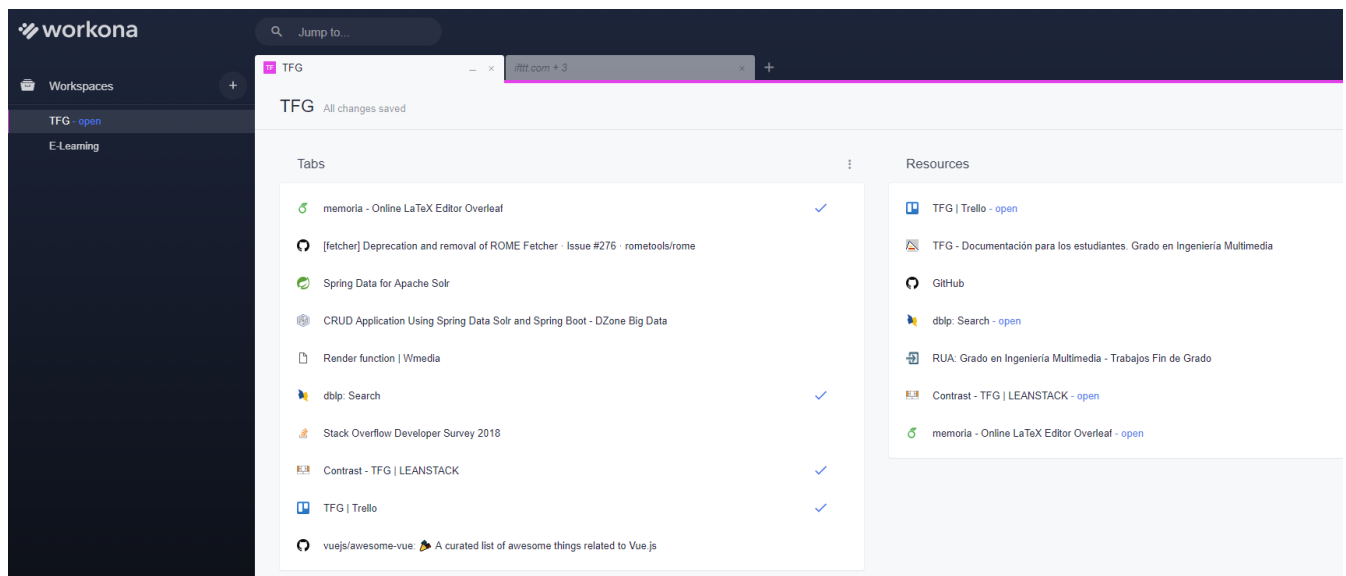


Figura 5.1.: Espacio de trabajo para Contrast

- Trello: aplicación web que posibilita la creación de tableros con listas de tareas. Dichas tareas pueden ser etiquetadas y comentadas y se les pueden añadir archivos adjuntos. Los tableros permiten el flujo libre de tareas entre una lista y otra. En el caso del proyecto, se ha utilizado una metodología de gestión de trabajo Kanban, por lo tanto se han creado tres listas: por hacer, en proceso y finalizado. Se han añadido dos listas adicionales: una para acumular las tareas al finalizar una iteración y otra para definir tareas que surgen durante el proceso de desarrollo y así asignarlas en una iteración próxima. <https://trello.com/>
- Todoist: aplicación web que facilita la organización de tareas en el día a día. En el caso del proyecto es utilizada para organizar las tareas personales junto a las del trabajo de final de grado. Esta aplicación es utilizada junto a IFTT que posibilita la integración de Todoist y Trello, de tal manera que cada vez que se cree una tarjeta en Trello, se añade una tarea en Todoist. La aplicación es utilizada bajo la filosofía GTD (Getting Things Done).



Figura 5.2.: Integración IFTT de Todoist con Trello

6. Análisis y especificación

6.1. Especificación requerimiento funcionales

6.1.1. RF01: Obtener un listado de noticias recientes

La aplicación debe ser capaz de mostrar un listado de noticias del día de uno o distintos medios agrupadas según la relación entre ellas al realizar la carga inicial de la página. Este es el comportamiento por defecto y se mantendrá y repetirá siempre y cuando el usuario no haya especificado un término de búsqueda.

6.1.2. RF02: Obtener un listado de noticias al realizar una búsqueda básica

La aplicación debe ser capaz de mostrar un listado de noticias de uno o distintos medios agrupadas según el hecho al que refieran al realizar una búsqueda. El término de búsqueda debe ser contrastado con el contenido, titular o categorías de las noticias para obtener un acierto.

6.1.3. RF03: Obtener un listado de noticias al realizar una búsqueda avanzada

La aplicación debe ser capaz de mostrar un listado de noticias de distintos medios agrupadas según el hecho al que refieran al realizar una búsqueda. Esta búsqueda debe especificar filtros avanzados, siendo estos las fechas en las que se contiene la noticia, los medios digitales en los que puede encontrarse y las categorías a las que pertenece.

6.1.4. RF04: Ampliar una noticia

La aplicación debe ser capaz de ofrecer una vista ampliada de una noticia que posibilite una lectura más sencilla y sin distracciones.

6.2. Casos de uso

7. Diseño

7.1. Arquitectura seleccionada

7.2. Tecnologías

7.3. Diagrama de clases

7.4. Mockups

8. Implementación

8.1. Iteraciones?

9. Pruebas y validación

10. Resultados

11. Conclusiones

11.1. Conclusiones

11.2. Líneas de trabajo futuras

Bibliografía

[AENOR, 1997] AENOR (1997). norma une 50136:1997.

A. Anexo I. Esquema de base de datos

Aquí vendría el anexo I

B. Anexo II. Otros...

Aquí vendría el anexo II