Sistemas de Información Orientados a Servicios

Trabajo final: Twico

Máster Universitario en Ingeniería Informática

diciembre de 2020

VNiVERSiDAD D SALAMANCA

Autores

Miguel Cabezas Puerto

Luis Blázquez Miñambres

Francisco Pinto Santos

Oscar Sánchez Juanes

En este documento se recoge la memoria desarrollada por los alumnos Miguel Cabezas Puerto, Luis Blázquez Miñambres , Francisco Pinto Santos y Óscar Sánchez Juanes del trabajo final de la asignatura “Sistemas de Información Orientados a Servicios” en el seno del Máster en Ingeniería Informática de la Universidad de Salamanca en el curso 2020-2021, consistente en una aplicación web para el tratamiento y gestión de temas relacionados con el virus SARS-COVID-19, presentando diversas tecnologías que ayuden a mostrar una perspectiva más general y simplificada de su implicación en el día a día de la sociedad..

Contenido

Índice de tablas

Índice de ilustraciones

1. Introducción

El ser humano, desde tiempos inmemoriales, ha buscado su expansión tanto física como de conocimientos. Así desde los primeros homínidos que realizaban armas arrojadizas para conseguir su comida sin necesidad de acercarse a una presa peligrosa hasta las actuales redes neuronales capaces de predecir decisiones humanas futuras, pasando por el descubrimiento de fuego, la invención de la rueda, avances en medicina como los trasplantes o la penicilina o, sobre todo en nuestro campo, la salida al mundo de Internet, el ser humano ha buscado una automatización de sus tareas para que estas le supongan el menor esfuerzo y coste posible, llegando incluso a poder disfrutarlas sin realizar más que un clic desde cualquier parte del mundo. En definitiva, el ser humano ha pasado de ser conocido como "social por naturaleza" a "globalizado por naturaleza", pudiendo disfrutar de todo aquello que desee, de la región del mundo que desee, desde el sofá de su casa.

Todo ello entra en contraposición con el frenazo en seco producido por la situación de pandemia actual causada por el virus SARS-COV-2 causante de la enfermedad COVID-19 [1]. Esta, de la noche a la mañana, ha sembrado el caos en el mundo obligando a cerrar fronteras entre países, distanciando socialmente a amigos y familiares, provocando decenas de miles de muertes en exceso, generando miedos e inseguridades en el conjunto de la población y, en definitiva, rompiendo teóricamente con aquel mundo tan globalizado previamente conocido.

No obstante, esta ruptura de la globalización ha resultado ser únicamente teórica y casi imperceptible en duración ya que, gracias a la inmersión que el mundo venía experimentando en la era digital, se han cambiado rápidamente las costumbres, incrementándose, más si cabe, la antigua globalización. Esto lo podemos ver en las nuevas "tele-quedadas", videoconferencias continuas, fiestas online… Y no es para menos ya que, como se ha comentado al inicio del presente documento, el hombre es un "ser social por naturaleza" necesita de las opiniones, reacciones e informaciones dadas por sus iguales para poder subsistir. Ello nos lleva de nuevo a la introducción, una vez más el hombre se ha expandido, esta vez de una forma digital.

Es por esta necesidad humana imperiosa de una diferente interacción, conocimiento del parecer de los demás y estar continuamente informado de los temas candentes de la actualidad, en este caso el coronavirus (pruebas, avances médicos, casos diarios locales, regionales, nacionales…) y sus consecuencias (políticas, económicas, sociales culturales…) que nace nuestro sistema, Twico, con los objetivos posteriormente detallados.

2. Objetivos del proyecto

Para el cumplimiento de las tareas y tecnologías planteadas dentro de la arquitectura del sistema, se marcaron una serie de objetivos a realizar para llegar a un producto que pudiera ser mantenido con el paso del tiempo y que fuera de utilidad, aplicando las diversas tecnologías que se explicarán posteriormente y con las cuales , los miembros del equipo del proyecto trabajan y colaboran de forma asidua.

En primer lugar, dada la actual situación de pandemia ya mencionada en la introducción, resulta de suma importancia monitorizar, controlar y gestionar el número de casos positivos, así como la cantidad de pruebas diagnósticas realizadas. Es por ello por lo que nuestro primer objetivo es recuperar, tratar y mostrar de una forma visual sencilla todos estos datos, con su respectiva diferenciación, tanto en Barcelona, como muestra de megalópolis española, como en otras regiones del mundo. Con ello las personas, de un simple vistazo, podrán enterarse de la cantidad de casos y pruebas realizadas en su región y así valorar qué medidas, a mayores de las exigidas, debe o quiere tomar (limitar sus contactos, evitar cenas en época navideña con familiares y allegados, cancelar o realizar visitas turísticas a dichas regiones…)

En segundo lugar, obedeciendo a la máxima presentada en la introducción de "el ser humano es un ser social y globalizado por naturaleza", las personas necesitan estar al día no solo de la actualidad en diversos ámbitos (político, sanitario, social…) sino conocer las reacciones y opiniones que el panorama nacional e internacional despierta en la población. Es por ello, unido a la preocupación general por el coronavirus, que nuestro segundo objetivo es recuperar, analizar y mostrar los *tweets* asociados a la temática COVID junto con otros *topics* deseados por el cliente de nuestro sistema indicando además el grado de positividad de lo escrito en ellos. Con ello pretendemos que el usuario logre el propósito de encontrar reacciones al tema central del sistema, COVID-19, sin tener que invertir excesivo tiempo en su búsqueda.

Unido a esto nace nuestro tercer y último objetivo, mostrar un resumen de noticias de medios digitales asociadas a los *topics* previamente mencionados. Con ello se pretende ahorrar tiempo al usuario en su búsqueda y lectura a la par que contrarrestar esta información con la obtenida a través de las reacciones de los internautas vía Twitter respecto de la misma temática, obteniendo finalmente la información en dos perspectivas, por una parte, una más objetiva por parte de la prensa y por otra, una más personal, subjetiva e interpretable, parametrizada en positiva o negativa, aportada por la comunidad de Twitter.

3. Modelado del sistema

En este apartado se definirá , de forma más abstracta, el diseño y planteamiento de la arquitectura del sistema, explicando en detalle los componentes que lo conforman, diferentes variables a tener en cuenta para su disposición y despliegue, así como un listado con el conjunto de APIs públicas utilizadas con motivo de la asignatura y que han sido fuente de datos principal dentro de la lógica de negocio de la aplicación.

3.1. Arquitectura del sistema

Twico está conformado por tres subsistemas principales, como se puede ver en la Ilustración 1:

- API de extracción datos: se trata del subsistema implementado en Python, expuesto a mediante un API Rest. Este alberga la funcionalidad para la extracción y formateado de datos a modo de ETL, siendo el componente principal ESB del sistema.
- Aplicación web: se trata de una aplicación web desarrollada sobre la pila MEVN+T, la cual está compuesta de dos subsistemas:
    - Página web: desarrollada en Vue.js, es la interfaz hombre-máquina que permite al usuario comunicarse con el sistema.
    - API servidor web: API desarrollada en TypeScript con node.js, que alberga las funcionalidades de servir la página web, autenticación del usuario con servicios externos y hacer de intermediario con el subsistema de extracción de datos.

En esta arquitectura se ha intentado mantener un servicio por cada subsistema, aislando así la lógica de extracción, análisis y tratamiento de datos en un subsistema separado de la aplicación web, aumentado así la reusabilidad de los distintos artefactos del sistema.

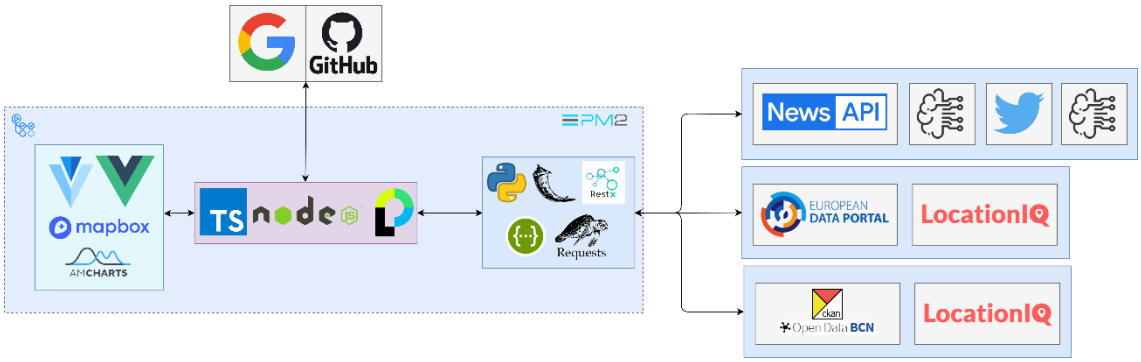

Ilustración 1: Arquitectura del sistema

3.2. Disposición de ETLs

Para llegar a cumplir los objetivos planteados en la sección Objetivos del proyecto se han planteado tres ETL, en los cuales se realiza la extracción, análisis, formateado y procesamiento de los datos.

Antes de comenzar, cabe destacar que todos los datos obtenidos en los ETL del sistema de Twico, son tratados posteriormente a su extracción, para poderlos servir en un formato normalizarlos. Este es el caso de las coordenadas que se proveen en la proyección WGS84, las fechas que se proveen en el formato ISO 8601 o los nombres de los países que se proveen en el formato ISO 31661-1 alpha2 y alpha3.

a) ETLs de extracción de datos de COVID

Para la extracción de los datos relativos a la pandemia de la COVID19, se ha decidido dividir la lógica de obtención de los datos, en dos ETL:

- El primero, que se puede ver en la Ilustración 2, es utilizado para extraer la información de los casos y defunciones relativos a el COVID existentes en los barrios de Barcelona, Para ello la información se ha extraído de un portal público de datos como es OpenDataBCN [2] mediante el estándar CKAN. Tras lo cual se ha enriquecido con haciendo geolocalización inversa con el API de localtionIQ, permitiendo así obtener las coordenadas de cada barrio a partir de el nombre de este.
- El segundo ETL se puede ver en la Ilustración 3, y consiste en un ETL que presenta un funcionamiento similar al anterior. No obstante, este recoge información de los casos y defunciones relativos a la pandemia de la COVID a nivel mundial, del portal de datos públicos de la unión europea. Tras ello obtiene el código ISO Alpha-3 de los países obtenidos, para poder tratar los datos en un formato estándar.

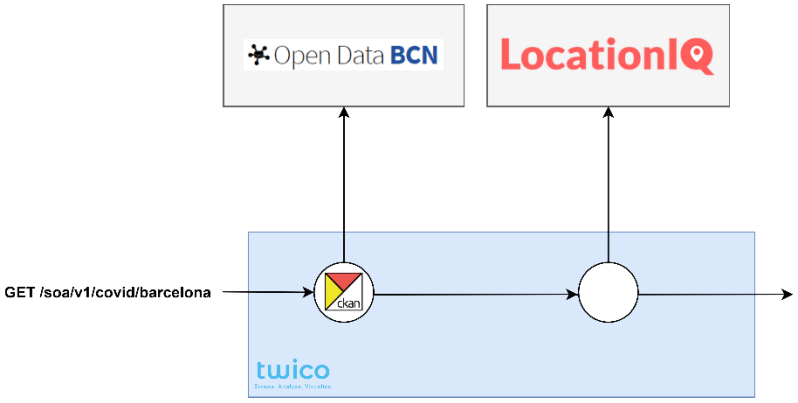

Ilustración 2: ETL para la extracción de información de COVID en Barcelona

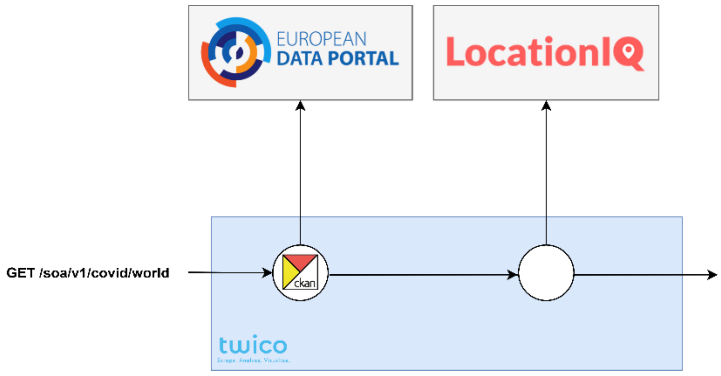

Ilustración 3: ETL para la extracción de información de COVID en el mundo

b) ETL de extracción de noticias, tweets y modelado de topics

Este EL es el más complejo del sistema, ya que presenta varias fuente de ingesta de carácter heterogéneo, además de dos análisis realizados mediante inteligencia artificial.

El proceso de obtención de datos de este ETL presenta los siguientes pasos:

- Extracción de noticias: mediante el API ofrecida por NewsAPI, extrae las ultimas noticias en ingles relativas a la COVID19.
- Tras ello se realiza un análisis LDA sobre el cuerpo de las noticias obtenidas, permitiendo obtener así los topics principales sobre los que se está hablando.
- Se toman los 5 topics más relevantes de los encontrados, y se realiza una búsqueda en Twitter de dichos topics junto a la palabra clave COVID, para encontrar información relativa a estos.
- Por último, se realiza un análisis de sentimiento sobre los tweets encontrados sobre cada topic, permitiendo conocer la neutralidad del texto escrito en estos y la opinión que se refleja sobre el topic buscado.

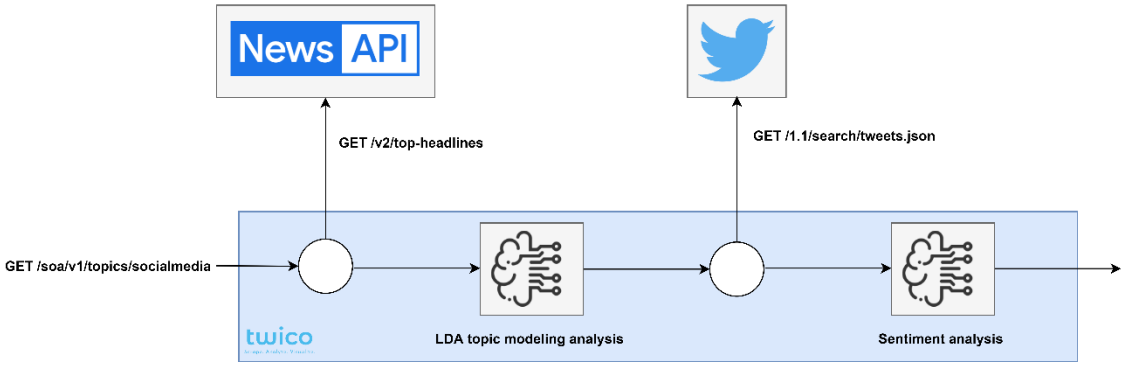

Ilustración 4: ETL para la extracción de información de noticias, tweets y topics

3.3. Uso de APIs externas

Para la gestión y funcionamiento del sistema se han hecho uso de diferentes APIs externas para la extracción de datos, todos acerca de la COVID-19. A continuación en la Tabla 1, se describirán cada una de las APIs utilizadas para el funcionamiento de la aplicación, el formato de los datos que devuelven y el objetivo de uso dentro del sistema.

Tabla 1: Tabla de APIs utilizadas

API	Endpoint	Descripción	Entrada	Salida
Twitter API	https://api.twitter.com/1.1/search/tweets.json	Devuelve la información referente a los datos de los tweets acerca de un topic o *query* dada	• **query**: palabra o conjunto de palabras por las que filtrar la búsqueda de tweets • **lang**: idioma de los tweets a recoger • **from**: fecha de inicio de búsqueda de tweets • **to**: fecha de fin de búsqueda tweets	Debido a la gran extensión del formato de salida de los datos, si se desea consultar el formato de salida de los datos de este endpoint acceder al siguiente enlace https://developer.twitter.com/en/docs/twitter-api/v1/tweets/search/api-reference/get-search-tweets
NewsAPI	http://newsapi.org/v2/top-headlines	Devuelve la información relativa las cabeceras de noticias asociadas a un país y a una categoría	• **country**: país de origen de las noticias • **category**: categoría de las noticias (deportes, salud, etc.) • **apiKey**: en formato texto la clave de la API para acceder a la información	Debido a la gran extensión del formato de salida de los datos, si se desea consultar el formato de salida de los datos de este endpoint acceder al siguiente enlace https://developer.twitter.com/en/docs/twitter-api/v1/tweets/search/api-reference/get-search-tweets
	http://newsapi.org/v2/everything	Devuelve los datos asociados a las noticias que contengan una palabra o conjunto de palabras de una búsqueda dada.	• **query**: topic con la palabra o conjunto de palabras para filtrar las noticias • **from**: fecha de inicio de búsqueda de noticias	Debido a la gran extensión del formato de salida de los datos, si se desea consultar el formato de salida de los datos de este endpoint acceder al siguiente enlace https://developer.twitter.com/en/docs/twitt

			• **to**: fecha de fin de búsqueda noticias • **sortBy**: punto de filtrado de las noticias (ascendente, descendente, …) • **apiKey**: en formato texto la clave de la API para acceder a la información	er-api/v1/tweets/search/api-reference/get-search-tweets
OpenData	https://opendata-ajuntament.barcelona.cat/data/api/action/datastore_search_sql	Devuelve toda la información de datos públicos acerca de la COVID-19 en la ciudad de Barcelona en formato encolumnado	• **sql:** cadena de texto que contiene la consulta en formato SQL sobre la búsqueda de datos públicos que se va a realizar	Debido a la gran extensión del formato de salida de los datos, si se desea consultar el formato de salida de los datos de este endpoint acceder al siguiente enlace: https://opendata-ajuntament.barcelona.cat/
	https://opendata.ecdc.europa.eu/covid19/casedistribution/csv	Devuelve toda la información de datos públicos acerca de la COVID-19 en Europa en formato encolumnado	Este endpoint no requiere parámetros de entrada para recoger los datos del CSV	Debido a la gran extensión del formato de salida de los datos, si se desea consultar el formato de salida de los datos de este endpoint acceder al siguiente enlace: https://www.ecdc.europa.eu/en/publications-data/download-todays-data-geographic-distribution-covid-19-cases-worldwide
LocationIQ	https://pypi.org/project/locationiq/	A través del endpoint “Forward geocoding” , recoge las coordenadas de latitud, longitud, altura, etc. y	Este endpoint no requiere parámetros de entrada para recoger los datos del CSV	Debido a la gran extensión del formato de salida de los datos, si se desea consultar el formato de salida de los datos de este endpoint acceder al siguiente enlace:

		toda la información acerca de una posición		https://docs.mapbox.com/help/tutorials/custom-markers-gl-js/

4. Conceptos teóricos

En este apartado se abordarán aquellos conceptos teóricos y explicaciones necesarias para entender ciertos aspectos del sistema desarrollado con un mayor nivel de comprensión y detalle. Estos conceptos aúnan explicaciones que no son de conocimiento general en el ámbito de la informática y , por ende, se explican a continuación con el fin de tener una base teórica a la hora de explicar conceptos posteriores.

4.1. ETL (Extract, Transform and Load)

Los procesos ETL (extraer, transformar y cargar, de sus siglas en inglés *Extract, Transform and Load*) son una importante parte de la integración de datos, sirviendo como un elemento importante dentro de las arquitecturas del sector de *Big Data* [3] cuya función completa el resultado de todo el desarrollo de la cohesión de aplicaciones y sistemas.

Es el proceso que permite a las organizaciones mover datos desde múltiples fuentes, reformatearlos y limpiarlos, cargarlos en otra base de datos para analizar, o en otro sistema operacional para apoyar un proceso de negocio.

ETL son las siglas de los pasos que conforman el proceso:

- **Extracción:** consiste en obtener datos de una o varias fuentes; estas pueden ser bases de datos, *APIs*, documentación, CSV (valores separados por comas, de sus siglas en inglés *Comma-Separated Value*), etc.
- **Transformación:** en este paso se realizan los cálculos, validaciones y limpieza de los datos obtenidos de la extracción, adecuándolos con el objetivo de obtener la información necesaria.
- **Carga:** los datos, una vez modificados para la obtención de información, se vuelcan en almacenes de datos, que al igual que en la extracción, puede ser de diferentes tipos: ficheros, bases de datos, *Data Warehouse*, etc.

Un ejemplo de la esquematización de un proceso *ETL* se puede observar en la Ilustración 5.

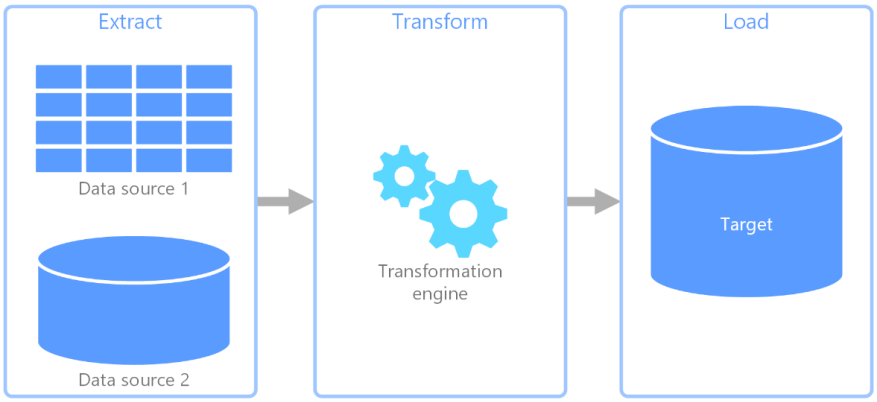

Ilustración 5: Esquema ETL

Este proceso ha sido utilizado en el proyecto debido a que permite obtener una gran cantidad de información de diversos medios, transformarla, adaptarla y almacenarla de forma que se pueda hacer uso posteriormente de estos datos [4].

4.2. Web Scraping

El concepto de *web scraping* basa su definición en dos pasos principales; la "búsqueda y descarga sistemática y automática de páginas webs y la extracción de información y contenido de estas". Cada vez que se mencione una fuente de datos, se estará haciendo referencia a la página webs o portal web de empleo de la que se extraerá la información.

El propósito de un *scraper* es la extracción de datos de las páginas webs a través de la "descarga" del contenido HTML (Hypertext Transfer Protocol) de la página web. Este proceso, normalmente automático, involucra la descarga del contenido de una página web (proceso realizado por un buscador cuando una página web es mostrada). Estas páginas webs pueden ser obtenidas como resultado del proceso de *web crawling*, para, posteriormente a su descarga, extraer la información adecuada para ser analizada, formateada, procesada o almacenada [5].

Las páginas web se crean utilizando lenguajes de marcado de texto (HTML y XHTML) y a menudo contienen una gran cantidad de datos útiles como texto. Sin embargo, la mayoría de las páginas web están diseñadas para usuarios finales humanos y no para facilitar el uso automatizado.

El uso particular de la técnica de *web scraping* es usada con propósitos de indexación, búsqueda web o *data mining*, entre otros.

4.3. NLP

También conocido por sus siglas *Natural Language Processing* (NLP) o Procesamiento del Lenguaje Natural, se trata de la práctica de comprensión acerca de cómo las personas organizan sus pensamientos, sentimientos, lenguaje y comportamiento para producir los resultados que obtienen. Este método proporciona a las personas una metodología para modelar un desarrollo personal y para el éxito de los negocios.

Un elemento clave de este campo de las ciencias de la computación, muy utilizado también en el campo de la lingüística y la inteligencia artificial, son las interacciones entre un ordenador y su capacidad de comprensión del lenguaje humano.

Esta práctica toma por referencia el lenguaje humano para aprender modelos computacionales útiles para la realización de determinadas tareas. Este campo cubre otros muchos conceptos dentro del procesamiento de lenguaje humano.

Un ejemplo de tipo de procesamiento de lenguaje natural sería el de NER o reconocimiento de entidades nombradas dentro de un texto. En el que, a través del aprendizaje y enriquecimiento del lenguaje humano, un sistema es capaz de detectar entidades relevantes o destacadas dentro de un texto con el fin de realizar análisis o estadísticas de clasificación más avanzadas.

Otro sería el campo de NLU o entendimiento del lenguaje natural. El cual, gracias a este concepto, permite valorar, identificar y clasificar un texto o una frase en base a su aproximación o cercanía a la forma de expresión de otros textos similares.

4.4. Algoritmo LDA

A partir del llamado *topic modelling* se consigue un proceso de identificar temas en un conjunto de documentos. Esto puede ser útil para los motores de búsqueda, la automatización del servicio al cliente y cualquier otra instancia en la que conocer los temas de los documentos

sea importante. Hay varios métodos para hacer esto, a continuación, se explicará uno de ellos: Latent Dirichlet Assignation (LDA).

LDA es un algoritmo de aprendizaje no supervisado que ve los documentos como bolsas de palabras o *bags of words* (es decir, el orden no importa). LDA funciona primero haciendo una suposición clave: la forma en que se generó un documento fue seleccionando un conjunto de temas y luego para cada tema eligiendo un conjunto de palabras. Para encontrar estos temas la respuesta se halla en aplicar ingeniería inversa a este proceso [6] **.**Para ello, los pasos a realizar para cada documento 'm' es el siguiente:

- Suponer que hay un número 'k' de temas en todos los documentos
- A continuación, se distribuyen estos 'k' temas en el documento 'm' (esta distribución se conoce como α y puede ser simétrica o asimétrica) asignando a cada palabra un tema.
- Para cada palabra 'w' en el documento 'm', hay que suponer que el tema asignado es incorrecto, y que al resto de las demás palabras se les asigna el tema correcto.
- Asignar probablemente la palabra 'w' a un tema basado en dos cosas:
    - ¿qué temas hay en el documento 'm'?
    - ¿cuántas veces a la palabra 'w' se le ha asignado un tema en particular en todos los documentos ¿(esta distribución se llama β)
- Finalmente, basta con repetir este proceso varias veces para cada documento y se formaría un conjunto o "bolsa" de palabras con los temas recogidos.

En cuanto a las variables que entran en funcionamiento en el algoritmo destacar las anteriormente mencionadas. Por un lado, α es una matriz donde cada fila es un documento y cada columna representa un tema. Un valor en la fila i y la columna j representa la probabilidad de que el documento i contenga el tema j. Una distribución simétrica significaría que cada tema se distribuye uniformemente en todo el documento, mientras que una distribución asimétrica favorece ciertos temas sobre otros. Esto afecta el punto de partida del modelo y se puede utilizar cuando tenga una idea aproximada de cómo se distribuyen los temas para mejorar los resultados.

Por otro lado, β es una matriz donde cada fila representa un tema y cada columna representa una palabra. Un valor en la fila i y la columna j representa la probabilidad de que el tema i contenga la palabra j. Por lo general, cada palabra se distribuye uniformemente a lo largo del tema, de modo que ningún tema esté sesgado hacia ciertas palabras. Sin embargo, esto puede aprovecharse para sesgar ciertos temas y favorecer ciertas palabras. Por ejemplo, si sabe que tiene un tema sobre los productos de Apple, puede ser útil sesgar palabras como "iphone" y "ipad" para uno de los temas con el fin de impulsar el modelo hacia la búsqueda de ese tema en particular.

A continuación, en la Ilustración 6, se muestra la imagen del modelo representado de forma más grafica del funcionamiento del algoritmo

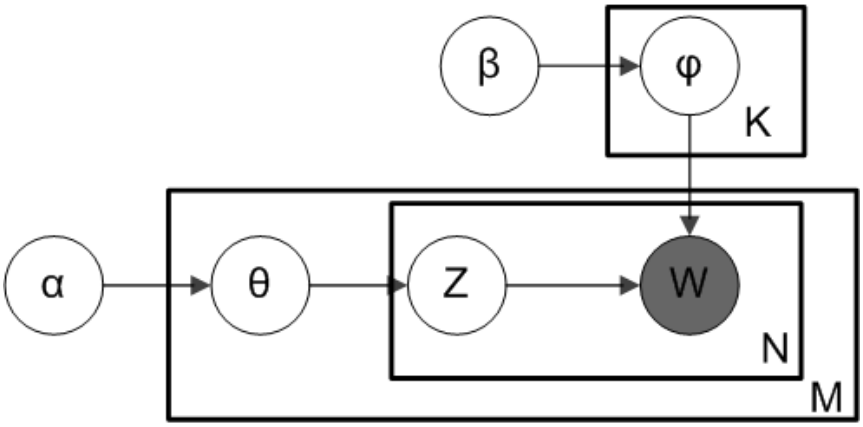

Ilustración 6: Modelo de funcionamiento del algoritmo LDA

5. Tecnologías y herramientas utilizadas

En este apartado se explicarán y detallarán aquellas herramientas y tecnologías utilizadas para el desarrollo de la aplicación y sus subsiguientes subsistemas, dando trasfondo y orientación de los conceptos explicados anteriormente y con una comprensión más detallada de forma individual de cada uno de los componentes de la arquitectura inicialmente planteada.

5.1. Lenguajes de programación

En este apartado se definirán y explicarán brevemente los lenguajes de programación utilizados en el desarrollo de la aplicación tanto de la parte de desarrollo web o *front-end* como de la parte de lógica de negocio o gestión de las APIs, *back-end*.

5.1.1. Python3

Ilustración 7: Logo de Python

Es un lenguaje versátil multiplataforma y multiparadigma que se destaca por su código legible y limpio. Es el principal lenguaje de programación utilizado en la parte de *backend* del proyecto, más concretamente, en el desarrollo del proceso ETL.

Python es ideal para trabajar con grandes volúmenes de datos ya que, el ser multiplataforma, favorece su extracción y procesamiento, es por esto una de las razones por las que se ha utilizado en este proyecto, con una parte dedicada al campo de *Big Data*, para la gestión de los volúmenes de datos extraídos acerca de ofertas de empleo y formación de los distintos portales web.

En el desarrollo de esta aplicación un factor importante debido a la implementación y ejecución de varios módulos dentro del proceso de extracción de datos. Los cuales, en cualquier otro lenguaje, supondrían un consumo de tiempo y de código programado mayor.

Entre algunas de sus múltiples ventajas, a continuación, se encuentran de manera resumida un conjunto de ellas [7]

- **Simplificado y rápido**: Este lenguaje simplifica mucho la programación, es un gran lenguaje para *scripting*.
- **Elegante y flexible**: El lenguaje ofrece muchas facilidades al programador al ser fácilmente legible e interpretable. Permite el tipado "dinámico" de variables, permitiendo declarar con seguridad una variable sin conocer su tipo de datos. Aunque esto, en ocasiones, puede resultar una desventaja si el código no está bien documentado.
- **Ordenado y limpio**: es muy legible y sus módulos están bien organizados.

- **Portable**: Es un lenguaje muy portable. Se puede usar en prácticamente cualquier sistema de la actualidad.
- **Comunidad**: Cuenta con un gran número de usuarios. Su comunidad participa activamente en el desarrollo del lenguaje.

5.1.2. Node.js

Ilustración 8: Logo de Node.js

A diferencia de Python, el otro lenguaje de programación importante que se ha utilizado en la aplicación es Node.js, un entorno de ejecución en JavaScript especializado en aplicaciones webs de tiempo real, servidores webs y en el manejo de recursos computacionales a alto nivel. Está basado en el motor JavaScript V8 de Google. Este motor está diseñado para ejecutarse en un navegador y ejecutar código JavaScript extremadamente rápido. La tecnología detrás de Node.js permite que este motor funcione en el lado del servidor, abriendo una gama completamente nueva de posibilidades cuando se trata del mundo del desarrollo [8].

Para trabajar de manera óptima, Node.js delega todo el trabajo a un grupo de subprocesos. Esta biblioteca tiene su propio entorno asincrónico de subprocesos múltiples. Node.js envía el trabajo a realizar al grupo.Node.js se diseñó teniendo en cuenta la escalabilidad, en particular con la capacidad de admitir una gran cantidad de conexiones simultáneas a un servidor [9]. Muchas tecnologías del lado del servidor ejecutan el entorno para cada una de las solicitudes en un hilo separado. Cuando aumenta el número de solicitudes, aumentan los recursos consumidos en el servidor. Además de los factores determinantes para el rendimiento de una computadora (RAM, CPU, velocidad de conexión), en un servidor las muchas veces el cuello de botella son los procesos de entradas y salidas (E/S).

Esto permite que, en el desarrollo de la aplicación web del proyecto actual, Node.js pueda manejar múltiples conexiones y solicitudes de manera muy eficiente, permitiendo soportar una cantidad considerable de conexiones simultáneas. Por el contrario, en general, Node.js no es adecuado en aplicaciones que requieren un número reducido de conexiones con un alto consumo de recursos (aplicaciones de cálculo, acceso intensivo a datos, etc.).

5.1.3. TypeScript

Ilustración 9: Logo de Typescript

TypeScript un lenguaje de código abierto que se basa en JavaScript, agregando definiciones de tipos estáticos. Estos tipos proporcionan una manera de describir la forma de un objeto, proporcionando mejor documentación y permitiendo a TypeScript validar que el código está funcionando correctamente [10].

El código TypeScript se transforma en código JavaScript a través del compilador de TypeScript o Babel.

5.2. Bibliotecas y frameworks

En este apartado se definirán y explicarán brevemente los conocidos *frameworks* y bibliotecas externas utilizadas durante el desarrollo del sistema y que ofrecen los lenguajes de programación Python o Node.js.

5.2.1. VueJS

Ilustración 10: Logo de Vue.js

Vue.js es un *framework* de JavaScript [11] enfocado a el desarrollo web. Concretamente es utilizado para construir interfaces web, basándose en el modelo *MVVM* (*Model*, *View*, *ViewModel)* [12].

La filosofía de desarrollo de Vue.js, se basa en construir componentes que sirve para realizar una pequeña tarea de control, visualización, o agregación de controles sobre la interfaz gráfica. Debido a ello, se dice que se basa en el modelo de ciclo de vida iterativo e incremental, permitiendo construir una página web de forma progresiva. [13]

En cuanto a Vuetify, es un pack de compontes para Vue.js, que aumenta el catálogo de componentes con tablas, botones, etc. para facilitar el desarrollo. [14]

En el desarrollo de nuestro servicio, en conjunto, se han empleado para la construcción de la página web y sus diferentes componentes como se explicará en próximas secciones.

5.2.2. VaderSentiment

VaderSentiment es una herramienta de código abierto bajo la licencia MIT[1] y cuyo propósito es el análisis de sentimiento basado en léxico y reglas [15].

[1] Licencia MIT: Es una licencia de software libre permisiva que solo impone restricciones muy limitadas a la reutilización y, por tanto, tiene una alta compatibilidad de licencias (es compatible con muchas licencias copyleft, como la Licencia Pública General GNU -GPL-).Permite la reutilización

Se caracteriza por estar en perfecta sintonía con la manera de expresar los sentimientos en redes sociales, de tal manera que es capaz de analizar correctamente oraciones complicadas que pueden hacer confundir a otras herramientas de este tipo. Además, trata adecuadamente el contenido multimedia (imágenes, vídeos…) adjunto a las publicaciones.

En cuanto a la evaluación de frases, cada oración tiene una puntuación final que determina su sentimiento. Esta puntuación se denomina puntuación compuesta y se calcula sumando las puntuaciones (un valor entre -1 -más negativo- y +1 -más positivo-) de cada palabra de la oración. De acuerdo con este criterio, una oración se considerará negativa si su puntuación compuesta es menor o igual a -0.05, neutra si se encuentra entre -0.05 y 0.05 y positiva si es mayor o igual a 0.05 [16].

Dentro del proyecto actual esta herramienta se usa para analizar cómo se siente la población con respecto a diversos temas relacionados con el COVID-19.

5.2.3. Scikit-learn

Scikit-learn es una de las librerías gratuitas que ofrece Python dentro del área de Data Science o ciencia de los datos para la composición y uso de algoritmos de clasificación, regresión, *clustering* y reducción de dimensionalidad, presentando la compatibilidad con otras librerías de Python como NumPy, SciPy y matplotlib.

La gran variedad de algoritmos y utilidades de Scikit-learn la convierten en la herramienta básica para empezar a programar y estructurar los sistemas de análisis datos y modelado estadístico. Estos se combinan y depuran con otras estructuras de datos y aplicaciones externas como Pandas [17].

Esta biblioteca se ha utilizado principalmente para la composición de un modelo mediante el algoritmo LDA, el cual se ha explicado anteriormente, que ha permitido la extracción de *topics* o temas de las noticias extraídas sobre la COVID-19 , con los cuales filtrar en la búsqueda de la API de Twitter aquellos tweets que hablasen del virus y de cada uno de los temas o *topics* extraídos.

5.2.4. Amcharts

Ilustración 11: Logo de amCharts

La biblioteca amCharts [18] permite la representación de información con base en JavaScript. Ofrece una filosofía de trabajo muy peculiar, distinta al del resto de bibliotecas de visualización para JavaScript, gracias a su variedad de gráficas básicas que incorpora y a sus diferentes opciones de configuración para cada una de ellas (personalización de colores, *tooltips*, formas, actualización, eventos o interacción, entre otros).

dentro del software propietario, siempre que todas las copias del software con licencia incluyan una copia de los términos de la licencia MIT y el aviso de derechos de autor.

El sistema utiliza esta biblioteca para mostrar los mapas con la información relativa al COVID-19 en el mundo y para mostrar gráficamente un resumen de los sentimientos relacionados con un determinado tema.

5.2.5. MapBox

Ilustración 12: Logo de MapBox

Mapbox [19] es una empresa emergente fundada en 2010. En menos de diez años, se ha convertido en una de las plataformas de mapas de código abierto más importantes del mundo. En 2015, la empresa logró una impresionante cifra de US $ 52,55 millones en una ronda de financiación.

Su éxito radica en proporcionar a los desarrolladores una gran cantidad de productos y servicios para diseñar mapas personalizados y crear aplicaciones basadas en sus herramientas.

En este artículo te explicaremos de forma clara y general qué productos y servicios podemos encontrar en Mapbox.

Actualmente Mapbox tiene 7 productos diferentes:

- Mapa
- Navegación
- Atlas
- Buscar
- Estudio de mapeado
- Visión
- Datos

Además, proporciona una serie de APIs y SDK para cada producto con el fin de obtener soluciones personalizadas. Y ha sido a través de sus APIs el método por el cual se ha accedido a los servicios que ofrece MapBox para el sistema de Twico, haciendo uso , específicamente, de los servicios de navegación y de mapas para representar los lugares de Barcelona y de Europa en los que se han detectado casos de COVID-19 a partir de los datos públicos extraídos mediante CKAN.

5.3. Herramientas de despliegue

En este apartado se definirán y explicarán brevemente otras herramientas de apoyo para el desarrollo de algunos de los subsistemas de la aplicación como pueden ser: despliegue automático, seguridad, extracción de datos, representación de gráficos, entre otros.

5.3.1. Git, GitHub y GitHub Actions

Git [20], es un software de control de versiones, creado por Linus Torvalds, centrándose en la simplicidad y eficiencia. Es utilizado para controlar los cambios que se llevan a cabo entre varios equipos, y es una herramienta de software libre bajo la licencia *GNU v2*. 37

GitHub [21] es una plataforma online de desarrollo de código en modo colaborativo, que trabaja con el control de versiones Git.

GitHub Actions [22] es el cliente de integración continúa ofrecido por el sitio web GitHub, que permite establecer una serie de instrucciones para realizar un despliegue, conocidas como *workflow*.

Estas serán ejecutadas cuando se dé una acción sobre una o un conjunto de ramas del repositorio, configurable en el propio *workflow*.

5.3.2. Certbot

Para habilitar HTTPS en un sitio web, es necesario obtener un certificado (un tipo de archivo) de una autoridad de certificación (AC o CA, su abreviatura). Entre la gran variedad de ACs que existen, Let's Encrypt es una CA con un alto reconocimiento. Con él, se podrá utilizar el software que utiliza el protocolo ACME que normalmente se ejecuta en el servidor web para realizar dicha acción de forma simplificada. [23]

Certbot es un cliente de automatización fácil de usar que puede obtener e implementar certificados SSL / TLS para un servidor web. Fue desarrollado por EFF y otras empresas como un cliente de Let's Encrypt, y anteriormente se conocía como "Let's Encrypt Official Client" o "Let's Encrypt Python Client". Certbot también cooperará con cualquier otra CA que apoye el acuerdo ACME.

Se recomienda que la mayoría de las personas con derechos de acceso a shell utilicen el cliente de ACME Certbot para realizar la emisión e instalación automática de certificados sin tiempo de inactividad, además de proporcionar un modo experto para aquellos administradores que no necesiten configuración automática. Es fácil de usar, se ejecuta en muchos sistemas operativos y tiene una extensa y detallada documentación.

En el caso del sistema de Twico, se ha utilizado para añadir e instalar los certificados SSL / TLS para el nombre de dominio del servidor adjunto de una forma fácil y flexible.

5.3.3. PM2

Se trata de un gestor de procesos *daemon* del sistema de GNU/Linux que permite manejar y gestionar de forma sencilla aplicaciones y servicios a través de una interfaz CLI (siglas en inglés de *Command-Line Interface*) [24]. Sirve principalmente para lanzar aplicaciones de Node.js en segundo plano, aunque también sirve para lanzar procesos realizando otro tipo de tareas como la ejecución de *scripts* o código en otros lenguajes de programación como Python.

Se trata de una librería gratuita que se suele utilizar en el desarrollo de aplicaciones web capaz de aguantar cantidades enormes de tráfico con un consumo de recursos realmente reducido y con herramientas que permiten realizar la monitorización de las aplicaciones de manera remota [25].

A través de PM2 se puede controlar un conjunto de procesos listados, que se arrancarán nuevamente en caso de error, manteniéndose encendidos mientras la máquina permanezca encendida. Es decir, en el caso que uno de ellos se termine por cualquier motivo, si se lanza una excepción de error en el programa que haga que el proceso se acabe, PM2 lo iniciará de nuevo automáticamente.

También permite otras herramientas como la gestión de logs, las herramientas de monitorización, el proceso de observación de archivos para rearranque automático cuando el código de la aplicación cambia, las utilidades de despliegue mediante un fichero JSON, etc.

5.4. Integración con otros servicios

En este apartado se definirán y explicarán brevemente aquellas tecnologías que no se clasifican como herramientas de uso sino como servicios externos, y que han permitido integrarse para la extracción de datos y como parte de la infraestructura del sistema de despliegue.

5.4.1. CKAN

CKAN es una herramienta para crear sitios web de datos abiertos. (similar a un sistema de gestión de contenido como WordPress, pero para datos, no para páginas y publicaciones de blog). Ofreciendo soporte para gestionar y publicar la recopilación de datos, se trata de una herramienta utilizada por gobiernos nacionales y locales, institutos de investigación y otras organizaciones que recopilan grandes cantidades de datos [26].

Una vez que se publican los datos, los usuarios pueden usar su función de búsqueda multifacética para buscar y encontrar los datos que necesitan, y usar mapas, gráficos y tablas para obtener una vista previa de los datos, ya sean desarrolladores, periodistas, investigadores, organizaciones no gubernamentales o ciudadanos. Incluso sus propios empleados.

CKAN es un software de código abierto y tiene una comunidad de desarrolladores activa que se compromete a desarrollar y mantener su tecnología central, siendo modificada y ampliada por una comunidad de desarrolladores más grande, contribuyendo a la creciente biblioteca de extensiones CKAN.

Está construido con Python en el *back-end* y Javascript en el *front-end*, y usa el marco web de The Pylons y SQLAlchemy como su ORM. Su motor de base de datos es PostgreSQL y su búsqueda es compatible con SOLR. Además, tiene una arquitectura modular que le permite desarrollar extensiones para brindar otras funciones, como recolección o carga de datos.

CKAN utiliza su modelo interno para almacenar metadatos sobre diferentes registros y lo muestra en una interfaz web que permite a los usuarios navegar y buscar estos metadatos. También proporciona una potente API que le permite crear aplicaciones y servicios de terceros a su alrededor.

5.4.2. OAuth 2.0

OAuth2.0, o por sus siglas en inglés, *Open Authorization 2.0*, se trata de un protocolo de autenticación, utilizado como un estándar en la industria de la información [27].

Está diseñado para ser simple y versátil, proporcionando así flujos de autenticación específicos para web, aplicaciones de escritorio, dispositivos, etc.

De la forma en la que se ha diseñado el protocolo, se le permite al usuario compartir información a un tercero, mediante la autenticación de este con el servidor de OAuth2.0 sin necesidad de que el tercero conozca su identidad.

El funcionamiento del protocolo se describe a continuación:

- Cuando el usuario desea autenticarse, se lo indica al tercero (en este caso un servidor de aplicación, que consiste en el sistema de Twico).
  No es estrictamente necesario, pero en este paso, en servidores que implementan acceso a información del usuario de distinta índole. En esta redirección, se envían también los ámbitos o *scopes*, que indican qué información va a solicitar más tarde el tercero (servidor de aplicación), para notificárselos a el usuario en la pantalla de confirmación.

- La aplicación, redirige al usuario a la pantalla de permisos (o confirmación) del servidor de OAuth2.0, para que conceda acceso a la aplicación.
  En esta pantalla, si se han indicado *scopes* en el paso anterior, se le indican al usuario para que sea consciente de a qué datos está permitiendo el acceso.

- En el caso de que el usuario autorice el acceso en la pantalla de confirmación, se le redirige a un *endpoint* de *callback* en el servidor de aplicación. En dicha redirección, se encuentra un código de autorización.
  En caso contrario el usuario es redirigido a la página anterior y el flujo termina.

- Con dicho código de autorización, el servidor de aplicación solicita un *token*, para poder acceder a las API cuyo acceso está limitado mediante OAuth2.0.

- El servidor de OAuth2.0 valida el código de autorización y envía un *token* temporal a el servidor de aplicación.
  También, opcionalmente el servidor de OAuth2.0, puede ofrecer un *token* de renovación, para que cuando caduque el *token* anterior, el servidor de aplicación pueda obtener uno nuevo sin necesidad de realizar los tres primeros pasos.

- Por último, el servidor de aplicación realiza las llamadas necesarias, a las API que le ha concedido acceso el servidor de OAuth2.0, aportando el *token* recibido en el paso anterior, mediante el cual el servidor de OAuth2.0 puede validar si permitir o no la petición.

6. Manual de usuario

En este apartado se mostrará y explicará , mediante una serie de capturas, el funcionamiento de la aplicación web y del sistema planteado de la solución final del proyecto Twico, con el fin de que sirva como guía de utilidad y como manual principal para los usuarios ajenos a la aplicación que deseen proveer de sus servicios.

Al acceder al sistema, lo primero que se presenta es la pantalla principal que se puede apreciar en la Ilustración 13 ,en la que hay una ilustración de la arquitectura del sistema. Además, le permite al usuario iniciar sesión con su cuenta de GitHub y de Google mediante los botones de la esquina superior derecha. No obstante, el acceso al sistema esta limitado por una *whitelist* definida en la aplicación web.

Una vez iniciada sesión, el usuario tiene acceso a el dashboard, como se puede ver en la Ilustración 14 se muestran una serie de visualizaciones respectivas a la información extraída sobre la COVID19:

- Mapa de casos de COVID19 en el mundo
- Mapa de defunciones en el mundo debido a la COVID19
- Mapa de casos de COVID19 en la ciudad de Barcelona (por barrios)
- Mapa de defunciones en la ciudad de Barcelona (por barrios) debido a la COVID19
- Topics encontrados en las noticias relativas a la COVID 19
- Tweets encontrados en función de los topics relacionados con la COVID19
- Resumen de análisis de sentimiento de los Tweets asociados a cada topic
- Ultimas noticias publicadas con relación a la COVID19

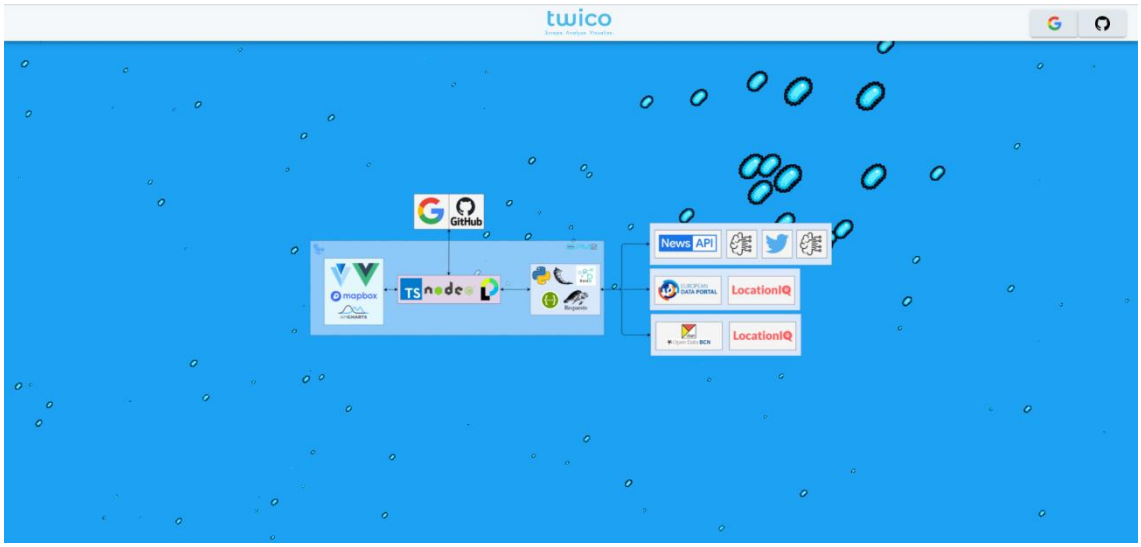

Ilustración 13: Vista de Inicio del sistema

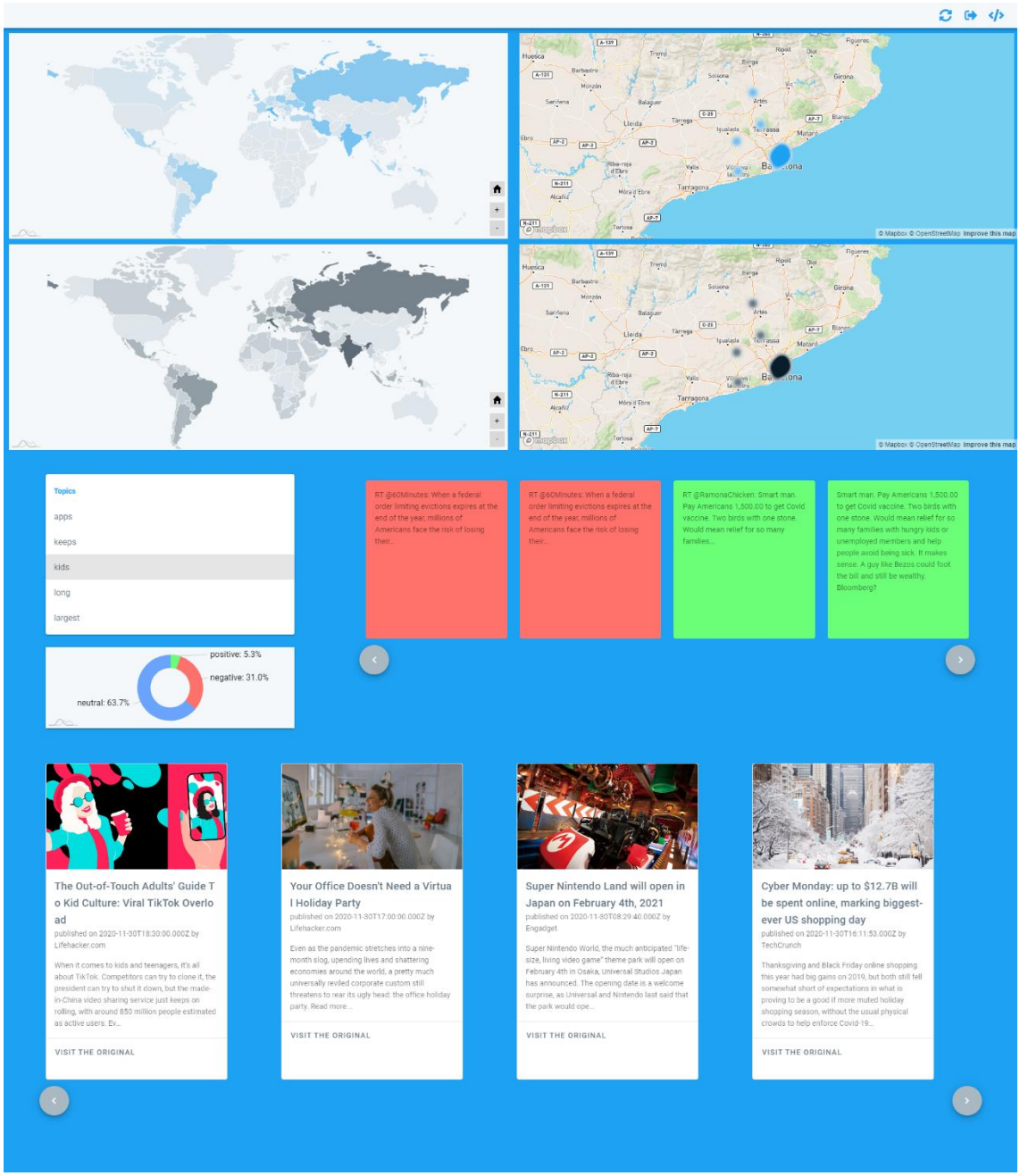

Ilustración 14: Dashboard del sistema

Bibliografía

[1] A. Trilla , M. Violan, J. M. Peri, E. Vieta Pascual y M. Rubinat, «Hospital Universitari Clínic Barcelona,» marzo 2020. [En línea]. Available: https://www.clinicbarcelona.org/asistencia/enfermedades/covid-19/definicion. [Último acceso: 18 diciembre 2020].

[2] «OpenData Barcelona,» [En línea]. Available: https://opendata-ajuntament.barcelona.cat/es/.

[3] O. B. Z. A. M. Bala, «Big-ETL: Extracting-Transforming-Loading Approach for Big Data,» Algeria, 2015.

[4] S. Pearlman, «Talend,» 19 Agosto 2019. [En línea]. Available: https://es.talend.com/resources/what-is-etl/. [Último acceso: 2020].

[5] D. K. M. a. L. Singh, "A dive into Web Scraper world," 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi: INDIACom, 2016, pp. 689-693.

[6] «TowardsScience,» [En línea]. Available: https://towardsdatascience.com/lda-topic-modeling-an-explanation-e184c90aadcd.

[7] «Ventajas y desventajas de Python,» Covantec, 2018. [En línea]. Available: https://entrenamiento-python-basico.readthedocs.io/es/latest/leccion1/ventajas_desventajas.html. [Último acceso: 2020].

[8] «Documentación de Node.js,» Node.js, [En línea]. Available: https://nodejs.org/es/docs/. [Último acceso: 2020].

[9] «Node.js: ¿Qué es y para que sirve NodeJS?,» Apasionados del marketing, 30 Septiembre 2015. [En línea]. Available: https://apasionados.es/blog/nodejs-4430/.

[10] «TypeScript Lang,» [En línea]. Available: https://www.typescriptlang.org/.

[11] MDN contributors, «MDN Web Docs,» 23 noviembre 2020. [En línea]. Available: https://developer.mozilla.org/es/docs/Web/JavaScript. [Último acceso: 18 diciembre 2020].

[12] J. Montero Ortega, «Open Webinars,» 4 febrero 2019. [En línea]. Available: https://openwebinars.net/blog/la-arquitectura-mvvm-y-sus-componentes/. [Último acceso: 18 diciembre 2020].

[13] «Vue.js,» [En línea]. Available: https://es.vuejs.org/index.html. [Último acceso: 18 diciembre 2020].

[14] «Vuetify,» [En línea]. Available: https://vuetifyjs.com/en/. [Último acceso: 18 diciembre 2020].

[15] «PyPi,» [En línea]. Available: https://pypi.org/project/vaderSentiment/.

[16] C. &. G. E. Hutto, A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. Eigth International Conference on Weblogs and Social Media, 2014.

[17] «Master Data,» [En línea]. Available: https://www.master-data-scientist.com/scikit-learn-data-science/.

[18] «amCharts,» [En línea]. Available: https://www.amcharts.com/.

[19] MapBox, «MapBox,» [En línea]. Available: https://www.mapbox.com/.

[20] Linus Torvald, «Git,» [En línea]. Available: https://git-scm.com/.

[21] GitHub, «GitHub,» [En línea]. Available: https://github.com/.

[22] GitHub, «GitHub,» [En línea]. Available: https://github.com/features/actions.

[23] Certbot, «Certbot,» [En línea]. Available: https://certbot.eff.org/.

[24] «PM2,» PM2, [En línea]. Available: https://pm2.keymetrics.io/docs/usage/pm2-doc-single-page/. [Último acceso: 2020].

[25] «Ejecutar una aplicación NodeJS en producción con PM2,» DesarrolloWeb, 6 Febrero 2020. [En línea]. Available: https://desarrolloweb.com/articulos/ejecutar-aplicacion-nodejs-pm2.html. [Último acceso: 2020].

[26] CKAN, «CKan,» [En línea]. Available: https://ckan.org/.

[27] Google, «OAuth 2.0,» [En línea]. Available: https://oauth.net/2/.