



UNIVERSIDAD DE JAÉN
Escuela Politécnica Superior de Jaén

Trabajo Fin de Grado

APLICACIÓN PARA EL ANÁLISIS DE TENDENCIAS EN TWITTER APLICANDO TÉCNICAS DE PLN

Alumno: Miguel Ángel González Gallardo

Tutor: Prof. D. Luis Alfonso Ureña López
Prof. Dña. Flor Miriam Plaza del Arco

Dpto: Informática

Junio, 2022



Universidad de Jaén
Escuela Politécnica Superior de Jaén
Departamento de Informática

Don Luis Alfonso Ureña López y Doña Flor Miriam Plaza Del Arco, tutores del Proyecto Fin de Carrera titulado: Aplicación para el análisis de tendencias en Twitter aplicando técnicas de PLN, que presenta Miguel Ángel González Gallardo autoriza su presentación para defensa y evaluación en la Escuela Politécnica Superior de Jaén.

Jaén, JUNIO de 2022

El alumno:

Miguel Ángel González Gallardo

Los tutores:

D. Luis Alfonso Ureña López

Dña. Flor Miriam Plaza Del Arco

Índice

1 INTRODUCCIÓN.....	5
1.1 Motivación.....	6
1.2 Objetivos del proyecto.....	7
1.3 Estructura del documento.....	8
1.4 Metodología.....	9
2 ANÁLISIS DE TENDENCIAS EN REDES SOCIALES.....	12
2.1 Redes sociales.....	12
2.1.1 Origen.....	12
2.1.2 Evolución.....	13
2.1.3 Twitter.....	15
2.2 Análisis en redes sociales.....	16
2.2.1 Introducción.....	16
2.2.2 Proceso de análisis.....	16
2.2.3 Estudio comparativo de plataformas.....	18
2.3 Procesamiento del Lenguaje Natural.....	23
2.3.1 Definición.....	23
2.3.2 Aplicación en el proyecto.....	24
3 ANÁLISIS.....	25
3.1 Especificación de Requisitos.....	25
3.1.1 Requisitos funcionales.....	25
3.1.2 Requisitos no funcionales.....	25
3.2 Planificación Temporal.....	26
3.2.1 Diagrama de Gantt.....	26
3.2.2 Diagrama PERT.....	31
3.3 Análisis de costes.....	31
3.3.1 Costes de desarrollo.....	31
3.3.2 Costes de funcionamiento.....	31
3.3.3 Cálculo de costes.....	33
4 DISEÑO.....	34
4.1 Casos de uso.....	34
4.1.1 Ver estadísticas de las tendencias actuales por región.....	34

4.1.2	Obtener estadísticas actuales a partir de una palabra clave.....	35
4.1.3	Crear Dataset (Conjunto de datos donde se van guardando los datos captados en tiempo real).....	36
4.1.4	Visualización, borrado y descarga de Datasets.....	37
4.2	Diagramas de secuencia.....	38
4.2.1	Ver estadísticas de las tendencias actuales por región.....	38
4.2.2	Obtener estadísticas actuales a partir de una palabra clave.....	39
4.2.3	Creación de un Dataset y visualización de sus estadísticas.....	40
5	PROPUESTA DE SOLUCIÓN.....	42
5.1	Arquitectura.....	42
5.1.1	Arquitectura Frontend.....	42
5.1.2	Arquitectura Backend.....	43
5.2	Interfaz.....	44
	Pantalla principal.....	44
	Listado de regiones para obtener sus tendencias.....	45
	Listado de tendencias de una región.....	45
	Búsqueda de tweets por palabra clave.....	46
	Estadísticas obtenidas a partir de región o palabras clave.....	46
	Vista de administración de Datasets.....	50
	Vista de las estadísticas de un dataset.....	53
6	IMPLEMENTACIÓN.....	54
6.1	Aplicación de SCRUM.....	54
6.1.1	Sprint 1.....	54
6.1.2	Sprint 2.....	55
	Ilustración 6.2. Icono de la aplicación.....	56
6.1.3	Sprint 3.....	56
6.1.4	Sprint 4.....	56
6.1.5	Sprint 5.....	57
6.2	Tecnologías.....	57
6.2.1	Python.....	57
6.2.2	Flask.....	57
6.2.3	API Google Maps.....	57
6.2.4	Tweepy.....	58
6.2.5	PySentimiento.....	58
6.2.6	Bootstrap.....	58
6.2.7	JQuery.....	58

6.2.8 DataTables.....	59
6.2.9 SweetAlerts.....	60
6.3 Herramientas.....	60
6.3.1 PyCharm IDE.....	60
6.3.2 Photoshop CS6.....	60
6.3.3 Gantt Project.....	60
7 PRUEBAS Y EVALUACIÓN DE USABILIDAD.....	61
7.1 Pruebas de validación.....	61
7.2 Pruebas de usabilidad y experiencia de usuario.....	62
8 CONCLUSIONES.....	66
8.1 Valoración personal.....	66
8.2 Posibilidades de mejora.....	66
ANEXO I – MANUAL DE INSTALACIÓN.....	68
ANEXO II – MANUAL DE USUARIO.....	69
ANEXO III – ÍNDICE DE ILUSTRACIONES.....	75
ANEXO IV – ÍNDICE DE TABLAS.....	77
Bibliografía.....	78

1 INTRODUCCIÓN

En este capítulo se presenta el trabajo de fin de grado. El objetivo del trabajo consiste en la investigación de los sistemas de análisis de tendencias en Twitter existentes y el desarrollo de un nuevo sistema.

Los sistemas de análisis de tendencias de Twitter, se basan en la recuperación de datos de Twitter a través de su propia API. Una vez recolectados estos datos, se computan y se generan una serie de estadísticas para que el usuario, o quizás, un modelo de inteligencia artificial (IA) pueda sacar sus conclusiones. El factor diferencial entre un sistema y otro es el grado de recuperación de información y el tipo de estadísticas que nos ofrece. Debido al gran tráfico de datos que maneja Twitter, su API no nos puede dar todo lo que se requiera, la API está limitada, por ello cada sistema de análisis de tendencias debe elegir qué datos quiere recuperar para no sobrepasar los recursos. Seguramente las estadísticas sea lo que más se tiene en cuenta a la hora de decantarse entre un sistema y otro, ya que no depende directamente de la API de Twitter, depende de la cantidad de recursos que dispongamos (CPU, tiempo de disponibilidad del sistema, peticiones de red por unidad de tiempo, memoria, almacenamiento...), si se requieren dar estadísticas complejas, el coste va a ser mayor.

El sistema desarrollado permite principalmente analizar tendencias actuales, generar conjuntos de datos con tweets en tiempo real y recuperar tweets por palabras clave o localización. Cabe destacar que la aplicación es multilingüe, concretamente podemos analizar completamente tweets en español e inglés. Las estadísticas que el sistema implementado nos ofrece son:

- Hashtags más usados.
- Usuarios más influyentes.
- Palabras más repetidas.
- Idiomas más usados por palabra clave o localización.
- Origen de tweets (Android, iPhone, Instagram...).
- Sentimiento y emociones de los usuarios que han publicado o retuiteado tweets.
- Detección de discurso de odio, si lo hubiera.
- Creación de datasets con la información recuperada en tiempo real.

1.1 Motivación

Tecnologías como Big Data, Minería Web, Procesamiento del Lenguaje Natural, etc. están viviendo su apogeo y, bien usadas, pueden ser un factor diferencial para cualquier empresa. Estas tecnologías tienen en común que necesitan grandes cantidades de datos para poder funcionar y qué mejor fuente de datos que las redes sociales, que ofrecen gran cantidad de información y actualizada.

Según los datos estadísticos [1] obtenidos en 2020 (Ilustración 1.1), el porcentaje de personas que usan las redes sociales en España según su franja de edad:

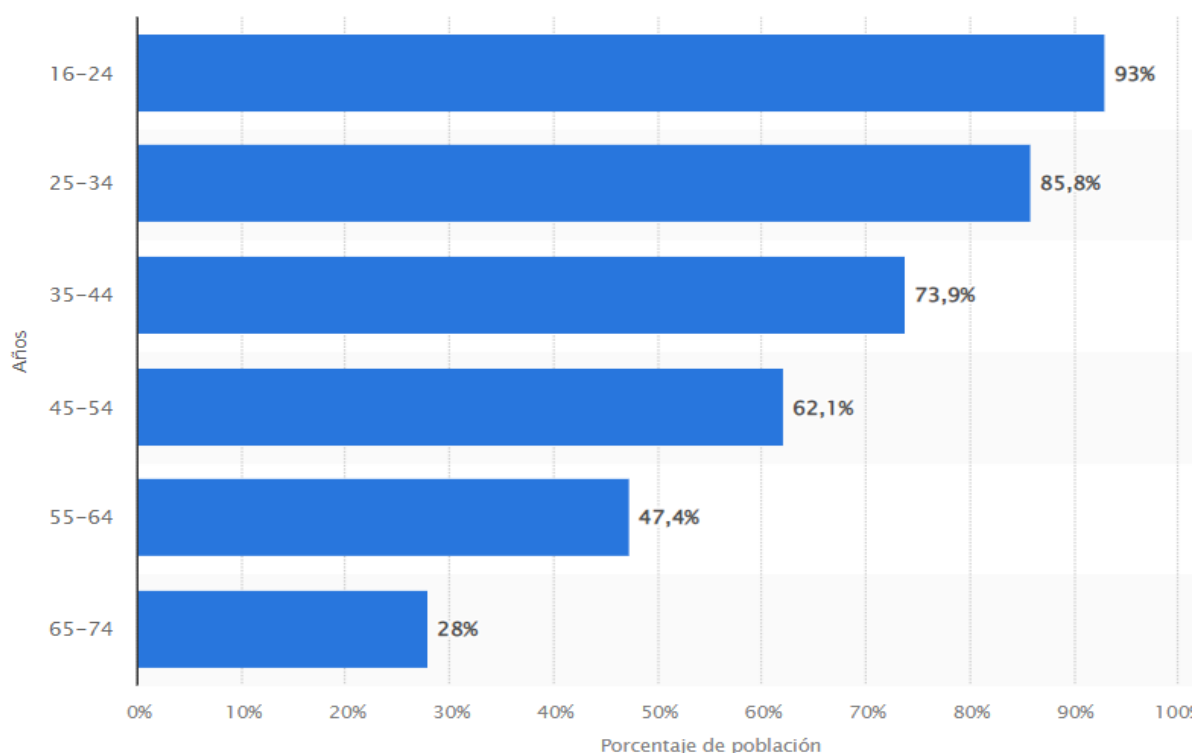


Ilustración 1.1. Usuarios RRSS por edad

Todas estas personas [2] [3] suponen una cantidad de 4 200 millones de personas de las que, de media, pasan 2 horas y 25 minutos diariamente en las redes sociales en todo el mundo. Como resultado de esto, tenemos una ingente cantidad de datos que se lleva generando durante años y que apenas se está comenzando a explotar.

Imaginemos que una empresa puede ver a tiempo real en qué lugares se está hablando de su producto y si hablan mal o bien. Todo esto prácticamente gratis comparando con los estudios de mercado tradicionales que se encargaban de analizar una muestra de población infinitamente más pequeña que la muestra de información que podemos obtener de las redes sociales. El empresario podrá actuar con tiempo y precisión para que su producto llegue a todos sitios y con una buena reputación, lo que se traducirá en ganancias para su empresa.

No obstante, no es solo monetizar. Por ejemplo, empresas como Facebook invierten en tecnologías para detectar comportamientos suicidas y evitar mostrar información que puedan “ayudar” a la persona a cometer finalmente suicidio.

A lo largo de la historia de las redes sociales, cada una ha ido posicionándose en su lugar de tal forma que podemos decir que Instagram se usa principalmente para subir fotos, TikTok para subir videos con un tono humorístico o incluso divulgación de conocimiento o productos, Facebook quizás sea una red social más personal donde subimos nuestras fotos familiares y algún que otro post, y por último Twitter, se ha convertido en la red donde uno tiene la sensación de opinar libremente y de que la gente lo escuche y se puedan reafirmar mediante un retweet.

Actualmente, debido al uso que hacemos de ella, Twitter sea seguramente la mejor red social para extraer información sobre la opinión y sentimiento de los usuarios. Además Twitter quizás sea una de las redes sociales que menos censura, esto permite la existencia de aspecto negativos como la propagación sin control de pornografía o imágenes de carácter sensible, pero también permite que las personas se puedan expresar libremente.

La necesidad de la implementación de nuevos sistemas de análisis de tendencias viene dada principalmente por las limitaciones de los existentes, hay pocos, no tienen apenas flexibilidad para la recuperación de los datos (no permiten personalizar la búsqueda de tweets) y no suelen incorporar técnicas de Procesamiento del Lenguaje Natural (PLN), y si lo incorporan, no suele ser multilingüe. La incorporación de técnicas de PLN ofrece grandes posibilidades a los sistemas que analizan tendencias ya que nos permiten conocer lo que los usuarios que usan las redes sociales han querido transmitir con su post (semántica, emociones, temas que se tratan, etc.), si no fuera por la inclusión de dichas técnicas en este proyecto, los resultados que se obtendrían perderían muchísima riqueza.

Los sistemas de análisis de tendencias que realmente pueden ofrecer soporte para una empresa son de pago, y no precisamente baratos, aunque las ganancias producidas por su buen uso lo justifican. Para el “público general”, existen sistemas gratuitos pero poco flexibles, con una funcionalidad limitada y con muchos anuncios.

1.2 Objetivos del proyecto

A continuación se exponen los objetivos a cumplir en la realización del proyecto y que servirán para evaluar todo aquello presentado en la memoria:

1. Realizar un estudio de las distintas plataformas disponibles actualmente sobre el análisis de tendencias en Twitter.
2. Realizar un estudio comparativo de las diferentes plataformas existentes para el análisis de tendencias en Twitter.
3. Aplicar distintas técnicas de PLN estudiadas durante el grado.
4. Desarrollar una aplicación web con el objetivo de visualizar y analizar las tendencias en Twitter

5. Redactar una memoria que recoja todo el trabajo desarrollado así como los manuales de instalación y usuario.

1.3 Estructura del documento

El presente documento está dividido en las secciones que se especifican y describen a continuación:

El actual y primer punto, “1. Introducción”, pone en contexto la necesidad de la implementación de sistemas de análisis de redes sociales, así como los objetivos necesarios y metodología realizada durante el desarrollo del proyecto.

El segundo punto “2. Análisis de tendencias en redes sociales”, describe con detenimiento las redes sociales, deteniéndose especialmente en Twitter ya que está fuertemente relacionada con el objetivo del proyecto. También explica el proceso de Análisis en redes sociales que hace referencia a distintas técnicas de análisis, fases del proceso de análisis y cómo un negocio podría explotar el análisis para sus propios intereses. Este proyecto a parte del análisis común de otras plataformas como son el seguimiento de los rankings de tendencias, usuarios, temáticas, etc. también usa técnicas de Procesamiento del Lenguaje Natural para obtener una mayor riqueza de los resultados obtenidos, por lo tanto en este punto también se define y se exponen las técnicas implementadas.

En el tercer punto “3. Análisis”, enumera los distintos requisitos funcionales y no funcionales que debe tener el proyecto. También se realiza una previsión de coste del proyecto en tiempo y en dinero.

En el cuarto punto “4. Diseño” identifica los distintos casos de uso y expone diagramas de secuencia para especificar de qué forma el sistema y el usuario va a interactuar y tener una idea más clara sobre la lógica a implementar.

En el quinto punto “5. Propuesta de solución”, está dividida en dos subsecciones que detallan la arquitectura software del backend de la aplicación y el storyboard con las distintas vistas que componen la aplicación, con una descripción de cada vista.

En el sexto punto “6. Implementación” se detalla el avance que se ha logrado en los distintos sprints desarrollados durante la implementación. También detalla y justifica las distintas tecnologías y herramientas que se usan o se han usado para el desarrollo del proyecto.

En el séptimo punto “7. Conclusiones”, se habla de las conclusiones sacadas tras la realización del proyecto y qué posibilidades habría si aumentamos el alcance del mismo.

Contiene cuatro anexos, el primero es un manual de instalación del software, el segundo un manual de uso de la aplicación, el tercero un índice de las distintas imágenes del documento y el cuarto y último anexo contiene un índice de las tablas del presente documento.

Para acabar se lista la bibliografía consultada para el desarrollo del proyecto.

1.4 Metodología

Como metodología se ha optado por una metodología ágil como SCRUM [4]. No se ha visto conveniente el uso de una metodología tradicional porque estas necesitan una especificación de requisitos iniciales muy precisos para que el resultado del proyecto sea satisfactorio. El proyecto se realiza desde una base de objetivos que deben alcanzar el proyecto, el problema es que los objetivos no son suficientes para obtener todos los requisitos. Entonces un enfoque iteracional (Ilustración 1.2) donde se vayan desarrollando requisitos y se vayan validando con los tutores, que actuarían como clientes, sería una solución más óptima.

La idea de SCRUM [9] es que se pueda descomponer todo el trabajo que supone el desarrollo de un proyecto en distintas actividades. Todas las actividades se irán definiendo y colocando en el “Backlog”.

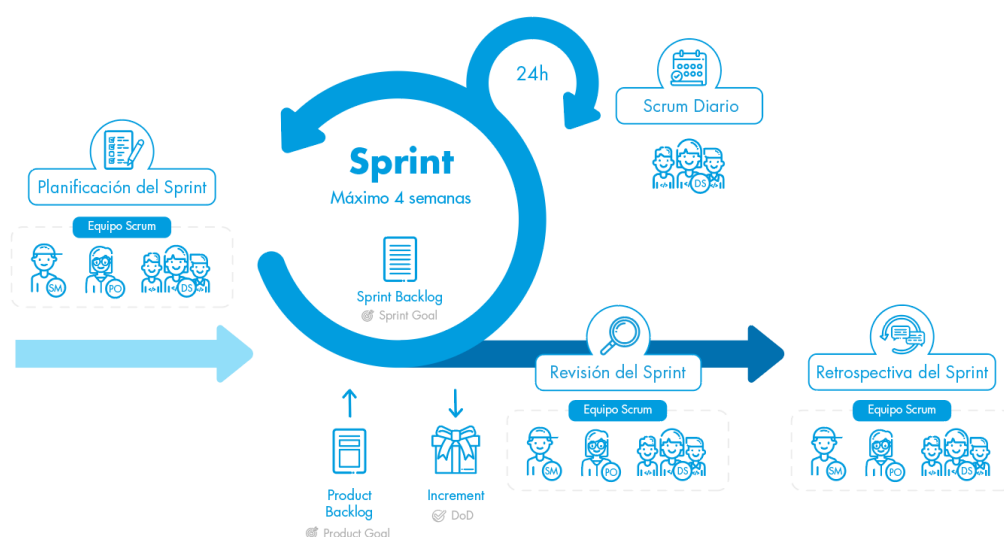


Ilustración 1.2. Ciclo de vida SCRUM

Esta metodología se basa en sprints, que son eventos de duración determinada que implican el desarrollo en equipo de una serie de actividades escogidas del Backlog y que una vez acabadas suponen la siguiente entrega del producto. Este proceso es un proceso iterativo que se va a repetir durante toda la vida de un software. Una vez terminado cada sprint se debe hacer una revisión con ayuda de las partes interesadas para inspeccionar resultados y determinar futuras adaptaciones, esto se hace como objetivo de aumentar la calidad y eficacia.

Antes, durante y después de cada sprint, son necesarias reuniones periódicas cuyo objetivo es conseguir transparencia, confianza y un mejor desarrollo en equipo.

Uno de los pilares en los que se basa SCRUM es la transparencia, todos las personas del equipo deben poder ver qué actividades surgen, cuáles están en desarrollo y cuáles están finalizadas para tener una idea clara del progreso del producto.

Se debe tener claro que SCRUM no es una metodología estricta y se debe adaptar lo antes posible a la forma de trabajar del equipo antes de que el producto resultante sea inaceptable. Se debe adaptar mediante un proceso de inspección realizada por alguna persona competente, aunque es vital que todas las personas involucradas posean capacidad para autogestionarse. El uso exitoso de SCRUM depende de cinco valores: Compromiso, enfoque, apertura, respeto y coraje.

Para la realización del proyecto, la principal adaptación de SCRUM viene dada en que el equipo se integra por alumno y tutores, y son estos últimos también los que validan las entregas realizadas por el alumno y proponen mejoras y modificaciones. La periodicidad de las reuniones tradicionalmente en SCRUM se mide en tiempo, pero ya que la disponibilidad y el tiempo dedicado por parte del alumno varía en el tiempo, se ha medido en hitos, cuando el alumno considera que ha conseguido un número de hitos que merezcan ser revisados, pide una reunión a los tutores para que les de el visto bueno o, en caso contrario, propongan una modificación. Para gestionar las tareas de una forma correcta se ha usado el tablero de GitLab¹ (Ilustración 1.3).

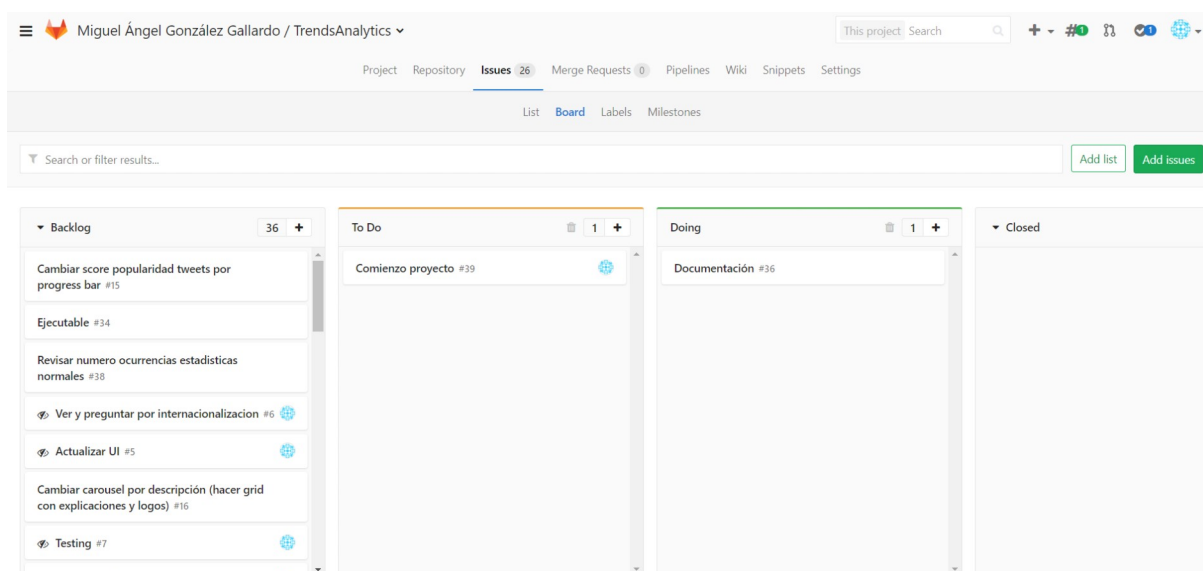


Ilustración 1.3. Tablero GitLab Issues

Como podemos ver, el tablero contiene cuatro columnas:

- Backlog. Actividades a realizar (pero no en el sprint actual).
- To Do. Actividades esperando a ser desarrolladas.
- Doing. Actividades en desarrollo.
- Closed. Actividades terminadas.

¹ <https://gitlab.ujaen.es/magg0034/TrendsAnalytics>

2 ANÁLISIS DE TENDENCIAS EN REDES SOCIALES

2.1 Redes sociales

2.1.1 Origen

Actualmente, la mayor parte de las personas conocen alguna red social, actualmente, las redes sociales más utilizadas en España son Facebook, Youtube, Instagram y Twitter. La primera red social parecida a las actuales es *sixdegrees.com*, que se remota a 1997. Esta red social permitía a los usuarios tener un perfil y una lista de amigos. Sixdegrees fue la primera en proponer un nuevo método de comunicación entre humanos, supuso el inicio de la explosión de redes sociales que vivimos hoy en día, su nombre se debe a la teoría de los seis grados, que afirma que una persona puede conectar con cualquier otra en tan solo seis pasos.

Inicialmente solo se estudiaban las redes sociales desde dos vertientes [5]:

- Las matemáticas, que usan la teoría de grafos para estudiarla: Los usuarios se representan como nodos y las relaciones entre ellos se representan como aristas. Las redes se representan mediante matrices, permitiendo un cómputo eficiente de las mismas.
- La psicología, inicialmente el contacto físico era primordial para las relaciones entre personas. A día de hoy, ya no lo es, y la juventud, como población más vulnerable está buscando en las redes sociales la inmediatez, la novedad y el mayor número de seguidores posible para satisfacer sus refuerzos sociales. Esto explica el exponencial incremento de número de usuarios en las redes sociales. Tampoco quiere decir que las redes sociales sean perjudiciales siempre y cuando su uso sea moderado y no tengan consecuencias psicológicas negativas.

El antropólogo Max Gluckman y su equipo de la universidad de Manchester unificó ambos aspectos y manifestó la necesidad de que eran necesarios nuevos conceptos y técnicas para analizar una red social compleja. Aparecieron términos como apertura, conectividad, proximidad, círculo social y densidad. Esto supuso las bases para generar nuevas técnicas de análisis de redes sociales, muchas de ellas usadas actualmente.

2.1.2 Evolución

SixDegrees fue un intento fallido, ya que cerró en 2001 pero sentó las bases para las siguientes redes sociales [10] [11]:

En 2002 llegaron Friendster, MySpace y LinkedIn, Friendster se enfocó en un red social enfocada a los videojuegos, mientras que MySpace y LinkedIn (actualmente con más de 600 millones de usuarios activos), se enfocan en un ámbito más profesional.

En 2004 Marc Zuckerberg creó Facemash, una red social que permitía opinar sobre las mujeres más y menos atractivas de la universidad de Harvard. Estuvo dos años existiendo solo para los alumnos de Harvard, no permitía registrarse en ella si no se tenía un correo institucional del centro. No fue hasta 2006 que Zuckerberg adaptó la red social a lo que hoy conocemos como Facebook. Al principio la red social estaba orientada a personas físicas las cuales creaban una lista de amigos con los que interactuar a través de publicar texto o contenido multimedia. Los amigos añadidos pueden interactuar con las publicaciones dando un “Me gusta”, comentándola o compartiéndola. La red social evolucionó, sobretodo cogiendo ideas de otras redes sociales que intentaron competir con ella, incorporaron historias (contenido multimedia disponible durante un tiempo para que lo puedan ver los amigos), grupos, páginas (orientadas a empresas u organizaciones), publicidad, etc. Gracias a la constante evolución de esta red social, sigue siendo la más utilizada con casi 3 000 millones de usuarios activos,

En 2005 surge Youtube, una red social originalmente destinada a un grupo reducido de amigos para compartir sus videos. La idea de la red social gustó bastante, los enlaces a los vídeos se compartían en otras redes sociales, lo que atraía más usuarios para subir contenido y consumirlo. Actualmente esta red social cuenta con 2.000 millones de usuarios activos al mes.

En 2006 un grupo de amigos crearon Twttr, un microblogging, donde cada usuario tenía 140 caracteres por post para expresarse. Esta red social supuso una revolución por su simplicidad y no tardó en evolucionar a lo que hoy conocemos como Twitter. A día de hoy se considera una red social esencial para que los medios de comunicación puedan conocer noticias, tendencias y consultar qué está pasando.

Con la aparición de los Smartphones y la proliferación de redes inalámbricas surgió en 2009 Whatsapp que reemplazó a los SMS, que hasta entonces era la principal forma de comunicación entre los jóvenes.

Instagram apareció en 2010, llegó al mercado como una red social enfocada a la fotografía. No tardó mucho en conseguir su fama gracias a compartir sus enlaces en redes sociales, sobretodo Twitter.

Twitch surge como una plataforma gamer en 2011. Actualmente es la única competencia real de Youtube ya que ha acaparado más del 75% de los streamings de videojuegos.

TikTok seguramente sea la última que se ha puesto de moda, surgió a finales de 2016. Permite compartir videos muy breves, así como editarlos con herramientas propias de la app.

A continuación se muestra una línea temporal del nacimiento de las principales redes sociales (Ilustración 2.1):

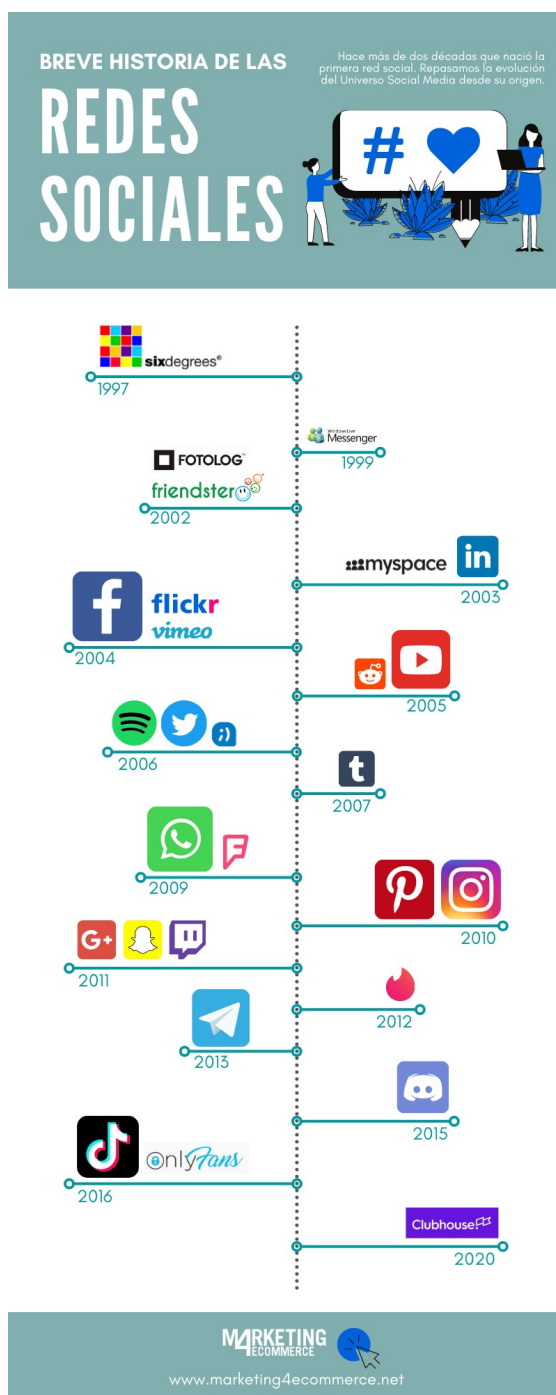


Ilustración 2.1. Genealogía redes sociales

2.1.3 Twitter

Se ha optado por Twitter como red social de la que obtener información acerca de las opiniones de los usuarios debido al perfil de los usuarios de la red social [12]. Según la Escuela de Negocios de la Innovación y los Emprendedores (IEBS), las redes sociales, en concreto Twitter, supone un factor diferencial a la hora de establecer relaciones de consumo con una marca. Las marcas cada vez dedican más esfuerzo a ofrecer servicio de atención al cliente en Twitter y, por parte de los usuarios, el 66% de los usuarios usan esta plataforma para informarse sobre un producto que están pensando en adquirir. Además, cuando se producen eventos, los medios de comunicación o el propio evento proporciona los Hashtags con los que expresar su opinión del evento.

Twitter [13], originalmente Twtr, es un servicio de microblogueo con sede en California. Se lanzó en Julio de 2006 y actualmente cuenta con más de 300 millones de usuarios que generan 65 millones de tuits diarios.

La idea básica de la red social es que los usuarios publiquen un breve texto (tuit), originalmente tenían un límite de 144 caracteres, pero a día de hoy ya podemos publicar un tuit con hasta 280. A los tuits le podemos dar retuit, que permiten agregar el tuit a nuestro tablón, aunque siempre citando al autor real. También podemos agregar a favoritos un tuit, que es una interacción parecida al retuit pero esta es menos visible por terceros.

Twitter dispone de una API² abierta para cualquier desarrollador que permite integrar Twitter en otras aplicaciones. Esta API permite hacer todo lo que podríamos hacer en la aplicación, como: Publicar tuits, mandar mensajes directos, leer tuits, etc. Esta API está limitada y sus limitaciones se ven modificadas en relación al tráfico que pueda Twitter soportar. Durante los años 2009-2013, Twitter tuvo serios problemas de estabilidad de su sistema, dando lugar al famoso error “Fail Whale”.

Algunas de las estadísticas [14] a nivel mundial y a 2022 más interesantes para la justificación del proyecto son:

- Cada día se publican 500 millones de tweets, por lo que se espera variedad de temáticas y de opiniones a analizar.
- Uno de cada cinco tweets, concretamente un 21.54 % de los tweets incluyen emoticonos. Estos emoticonos son muy útiles para entender lo que quiere expresar un usuario.
- Un usuario de twitter tiene, como promedio, 707 seguidores con los que interactuar.
- Más del 83% de los líderes mundiales están en Twitter. Por lo que podemos ver cómo de influyentes son respecto un tema.
- El 80% de los usuarios accede a través de dispositivos móviles, los que nos puede ofrecer bastante ventajas a la hora de obtener estadísticas con la geolocalización.

2 <https://www.tweepy.org/>

- El 52% de usuarios acceden a la plataforma todos los días, lo que nos dice que la mayoría de usuarios son constantes.
- Agencias como la CIA, analizan 5 millones de tweets al día. Los que nos da una orientación de cuán importantes son los analizadores de tendencias en la actualidad para los gobiernos.
- El 66% de las empresas con más de 100 empleados tienen cuenta de Twitter. De esas empresas, el 54% usan Twitter como herramienta de marketing digital.

2.2 Análisis en redes sociales

2.2.1 Introducción

El análisis de redes sociales (ARS), [6] [7] [15] es un campo de estudio interdisciplinario (ciencias sociales, matemáticas, estadística e informática), enfocado en el estudio de las redes sociales y cómo inciden estas en la sociedad.

El análisis de las redes sociales se creó mucho antes que las redes sociales que se conocen actualmente. Se remonta a 1924 con la creación de los sociogramas que consisten en la utilización de la teoría de grafos para el estudio de las estructuras sociales relacionales. Esto supuso la generación de formalismos y teorías que posteriormente, junto con avances en los campos de la psicología, antropología, estadística e informática, han sentado las bases para el análisis de las redes sociales que se aplica a día de hoy.

2.2.2 Proceso de análisis

Originalmente el análisis de redes sociales solo tenía en cuenta la estructura de la misma, es decir, los usuarios y las relaciones entre ellos. A día de hoy, se tiene más en cuenta el contenido de las interacciones entre los usuarios y es por ello que las técnicas de análisis que se aplicarán en este proyecto se limitan a la información de los tweets y no de la información de los usuarios. Antes de entrar en detalle tenemos que definir ciertos conceptos básicos para entender este tipo de análisis:

- **Actor:** Son unidades discretas de usuarios que componen una entidad. Podemos asociar erróneamente actor a usuario, no siempre es así, por ejemplo, un grupo o una página de Facebook componen un Actor pero está formada por varios usuarios.
- **Lazo relacional:** Es el medio por el cual se relacionan dos actores. Las relaciones pueden ser de muchos tipos:
 - Amistad.
 - Transferencia de información.
 - Interacciones.
 - Cercanía geográfica, en estatus social o en ocupación.

- Parentesco.
- Redes unimodales: Todos los actores son del mismo tipo. Por ejemplo, una red de trabajadores de una empresa.
- Redes bimodales: Se dan cuando podemos definir el problema que queremos analizar como dos conjuntos de actores, con al menos una relación entre ambos conjuntos y un conjunto de acontecimientos que ocurren en cada grupo. El objetivo de este análisis consiste en definir un conjunto de funciones que midan la afiliación entre un conjunto de actores y el otro.
- Redes multimodales: En este tipo de redes hay varios tipos de actores, en este tipo de redes agrupamos los actores en dos conjuntos, actores homogéneos, que comparte atributos, y actores heterogéneos.
- Redes egocéntricas: Este tipo de redes se da cuando un usuario está relacionado directamente con multitud de usuarios.

La teoría de grafos es la base fundamental para el análisis de la estructura de las redes sociales. La teoría de grafos ofrece una terminología muy compatible con las propiedades inherentes de las redes sociales, además de una base matemática que proporciona capacidad para demostrar teoremas y deducir enunciados. En informática usamos las matrices para representar los grafos, ya que contienen la misma información y su cómputo es muchos más eficiente.

Actualmente, el análisis de estructura, aunque se usa, no es el más usado para redes sociales. El análisis de contenido resulta mucho más interesante y versátil para las empresas y por ello nos centraremos en él en este trabajo. Podemos utilizar estas técnicas para:

- Clasificación de los mensajes por temática o categoría. Para ello se usa conjuntos de entrenamiento con los que entrenar modelos sean capaces de clasificar futuros mensajes en la categoría correspondiente con una buena precisión.

En estos casos, es importante elegir bien qué modelo vamos a usar, el conjunto de entrenamiento y la función con la que valoraremos cómo de bueno es un resultado.

A la hora de elegir el conjunto de entrenamiento debemos de intentar que este sea lo más variado posible y que no sea ni muy pequeño ni muy extenso, ya que el equilibrio de estos dos factores va a determinar la capacidad para reconocer y generalizar.

- Analizar la popularidad:

Para medir la popularidad de un post en una red social recurrimos al grado de “molestia” que se han tomado los usuarios para interactuar con el post, y es por ello que una respuesta vale el triple que un “favorito”. El favorito implica solo pulsar un botón, mientras que una respuesta le supone más tiempo al usuario para crearla. Podemos deducir que si muchos usuarios han invertido mucho tiempo en

interaccionar con el post, el post es más relevante. Las formulas expuestas son orientativas y pueden variar según el tipo de análisis:

- Twitter:

$$\frac{N^{\circ} \text{ de favoritos} + (2 \times N^{\circ} \text{ de retweets}) + (3 \times N^{\circ} \text{ de respuestas})}{N^{\circ} \text{ de seguidores}}$$

- Facebook:

$$\frac{N^{\circ} \text{ de me gusta} + (2 \times N^{\circ} \text{ de compartido}) + (3 \times N^{\circ} \text{ de comentarios})}{N^{\circ} \text{ de seguidores}}$$

2.2.3 Estudio comparativo de plataformas

Antes de empezar a diseñar la aplicación que compone parte del proyecto se ha hecho un estudio de las aplicaciones existentes en el mercado, a continuación se exponen algunas y se comentan sus principales características:

- [Hootsuite](#): Más que una aplicación en sí, es una herramienta que permite automatizar ciertas funciones de las redes sociales y obtener algunas estadísticas de nuestra actividad en múltiples cuentas que tengamos. Está destinada a Community Managers de empresas y a influencers. Básicamente con esta herramienta podemos automatizar los posts, orientarlos a un grupo de personas específico, medir el resultado de las publicaciones y supervisar a clientes, competidores y hashtag. Podemos ver que la tarifa de precios es relativamente costosa, estos precios van en función de la cantidad de opciones que te proporciona la herramienta. Dashboard de la aplicación (Ilustración 2.2):

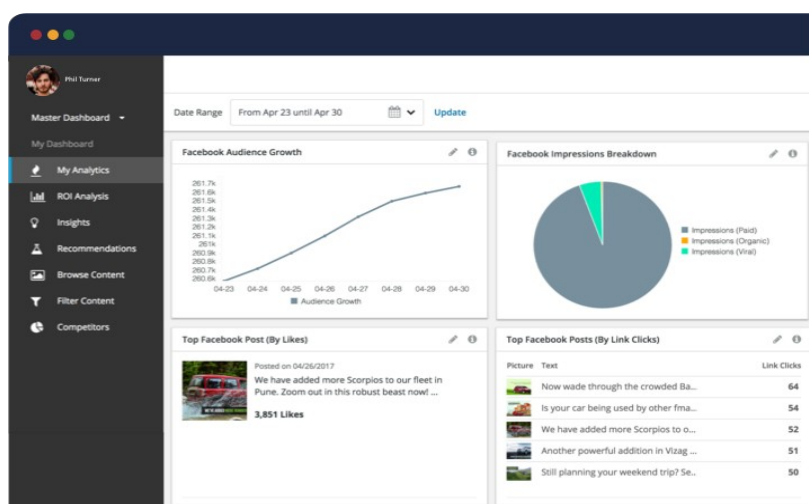


Ilustración 2.2. Dashboard herramienta de análisis HootSuite

- [Xovi Suite](#): Dentro de esta Suite de herramientas, existe la herramienta “Social Analytics”, con esta herramienta podemos ver el número de seguidores activos, ver de qué país provienen y cómo han influido en una tendencia. Podemos sacar

estadísticas de un perfil, post por hora, número de interacciones... También podemos hacer un seguimiento de los hashtags, enfocado principalmente para concursos que pueda hacer la empresa.

A diferencia de la Suite anterior, esta no está tan enfocada directamente en las redes sociales, está más bien enfocada en el posicionamiento web, es decir, se emplea para lograr que una web tenga más relevancia. Para ello, una de las opciones que permite es asociar las cuentas de redes sociales de la empresa, una vez asociadas, existe la herramientas de monitorización.

Cuenta con una herramientas interesantes que nos permiten elegir qué palabras clave utilizar en nuestros post para llegar a más gente, analizar los links de las publicaciones recuperadas, analizar por idioma y/o lugar, ver cómo de visibles son sus anuncios...

- [GetDayTrends](#): En este sitio web podemos ver las tendencias por zona geográfica. Lo interesante de esta herramienta es que nos permite ver la evolución de la tendencia a lo largo del tiempo (Ilustración 2.4), además permite ver desde qué lugares se publicó tweets que pertenezcan a la tendencia. A parte de esto, no ofrece mucha más nformación.

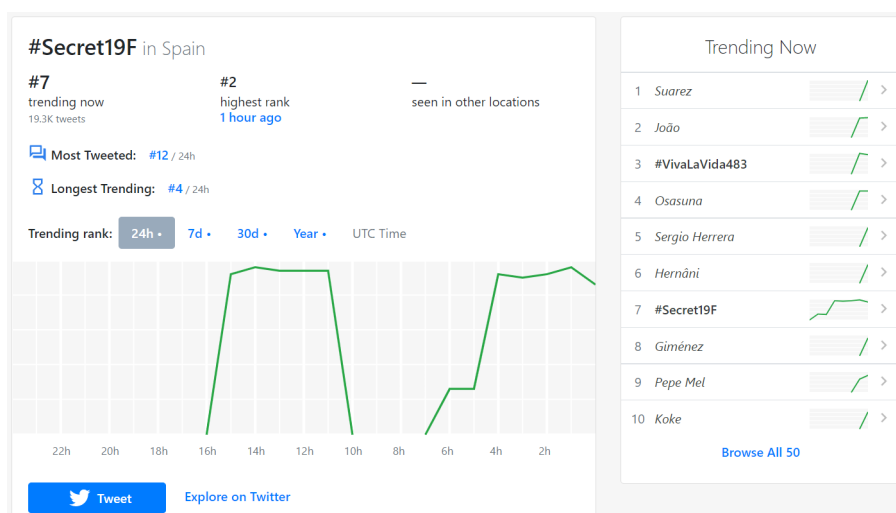


Ilustración 2.4. GetDayTrends evolución de tendencia

- [Tweeplers](#): Es un sitio web de acceso gratuito, completo, y sin mucha publicidad, una de las mejores opciones para una búsqueda rápida. Con esta herramienta podemos ver las personas de las que más se está hablando por país, lista de hashtags (Ilustración 2.5) más usados por país, lista de países que más twitteen... Además podemos ver las estadísticas en un mapa (Ilustración 2.6), todo a tiempo real:

SEMRUSH		
Domain, keyword or url		
Rank	Hashtag	Tweets /2K
Ad	#Ultimate_20_in_1_SEO_Tool	--
1	#Madrid	12
2	#VivaLaVida483	10
3	#Sevilla	9
4	#CopaACB	9
5	#Barcelona	8

Ilustración 2.5. Tweepplers Hashtags List

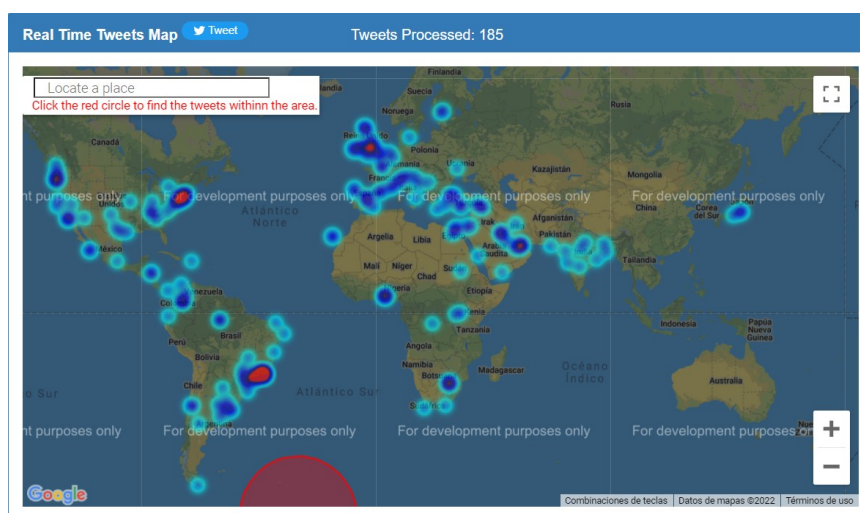


Ilustración 2.6. Tweepplers Real Time Tweets Map

Para facilitar la comprensión de la comparativa entre las aplicaciones estudiadas se ha elaborado una tabla (Tabla 2.1):

	Hootsuite	Xovi Suite	GetDayTrends	Tweeplers
Gratuito	No.	No.	Sí.	Sí.
Permite analizar varias RRSS	Sí.	Sí.	No.	No.
Permite personalizar la búsquedas	Si.	Si.	No.	No.
Soporte	Sí.	Sí.	No.	No.
Visualiza tendencias por país	No.	No.	Sí.	Sí.
Incorpora técnicas de PLN	No.	No.	No.	No.
Permite guardar información de las búsquedas	Sí.	Sí.	Sí.	Sí.
Permite recuperar tweets por palabras clave	Sí.	Sí.	No	No

2.1. Tabla: Comparativa de analizadores de tendencias

Tras ver aplicaciones similares a la que se desarrolla en este proyecto, he concluido que las aplicaciones existentes que realmente son de utilidad para empresas son de pago, y además bastante caras, solo orientado para empresas medianas y grandes. Las aplicaciones gratuitas solo permiten unas pocas opciones, no podemos personalizar las búsquedas y monetizan con mucha publicidad.

Para el desarrollo del proyecto me he basado en las aplicaciones vistas, haciendo un híbrido entre las de pago y las gratuitas, ya que el alcance del proyecto no es tan grande como para implementar un suite de herramientas de marketing digital y las posibilidades que ofrecen los analizadores de tendencias gratuitos se consideran poco útiles.

Además, la aplicación desarrollada incorpora grandes novedades gracias a la inclusión de técnicas de PLN que nos permitirán que la aplicación sea capaz de entender lo

que el usuario ha querido expresar con su tweet, y una vez comprendido, sacar conclusiones y estadísticas.

A todo esto se le suma la generación de corpus y que la aplicación permite el análisis completo de tweets en español e inglés, detectando los sentimientos, emociones y lenguaje del odio, si lo hubiese.

Que una aplicación de este tipo incorpore técnicas de PLN supone una enorme ventaja respecto a aplicaciones similares ya que, los tweets, no dejan de ser un texto del que prácticamente no conocemos nada, las palabras y hashtags más repetidos nos pueden dar una idea del contexto de una tendencia, pero gracias al PLN podemos ir más allá, podemos conocer la semántica de cada tweet y obtener unas estadísticas más precisas que no podríamos obtener de otra forma.

2.3 Procesamiento del Lenguaje Natural

2.3.1 Definición

El PLN, en inglés, *NLP (Natural Language Processing)*, es un campo de la IA que, junto a la lingüística, tratan de que los sistemas computacionales sean capaces de comprender textos y palabras habladas de la misma forma que los seres humanos. Cada vez se aplican técnicas de PLN a más campos pero entre sus aplicaciones típicas nos podemos encontrar:

- Obtener resúmenes automáticos de un texto.
- Traducir textos.
- Reconocimiento de entidades (Las entidades pueden ser personas, organizaciones, lugares, etc.).
- Análisis de sentimientos.
- Reconocimiento de voz.
- Clasificación por temática.

Para este proyecto en concreto se ha utilizado técnicas de análisis de la polaridad que permite saber si el texto de un tweet presenta connotaciones positivas o negativas, análisis de emociones [17] que permite extraer de manera automática y mediante técnicas computacionales información subjetiva expresada en texto, y detección del lenguaje del odio, que permite detectar si existen expresiones ofensivas en un texto.

La inclusión de PLN en este tipo de proyectos nos proporciona técnicas capaces de obtener información de lo que los usuarios quieren transmitir con sus tweets, de tal forma que podemos obtener unas estadísticas más precisas e importantes. Entre otras cosas, podemos obtener información sobre las temáticas de los tweets, podemos calcular en nivel de agrado o desagrado que el usuario pueda expresar en sus tweets, podemos detectar

odio en los tweets e incluso podemos reconocer qué entidades están involucradas en un tweet.

2.3.2 Aplicación en el proyecto

La recuperación de tweets puede ser en directo (streaming) o tweets del pasado. Una vez recuperados los tweets, debemos realizar un preprocesamiento para no tener en cuenta palabras como “de”, “a”, “por”, que son palabras que usamos mucho en español y que si hacemos un análisis con ellas podemos interpretar que son palabras relevantes, y no lo son. A estas palabras las denominamos palabras vacías o stopwords, son aquellas palabras que carecen de significado por sí solas.

Podemos dividir el análisis que le hacemos a los tuits que obtendremos con la API en dos tipos de análisis:

1. **Análisis básico:** En este análisis haremos un preprocesamiento simple que consiste en quitar URLs, quitar palabras como “RT” y eliminar palabras vacías. Como es lógico, el conjunto de stopwords es dependiente del idioma, por ello he utilizado las stopwords del corpus de NLTK, con esta herramienta tendremos las stopwords de los principales idiomas.

Una vez preprocesado el texto, iteramos instancia por instancia, haciendo un recuento de los hashtags (palabras que empiezan por #), usuarios (palabras que empiezan por @) y un recuento de palabras que no sean ni hashtags ni usuarios.

De la misma forma se hace un recuento de los idiomas de los tuits y de los dispositivos desde donde se han publicado.

2. **Análisis de sentimientos:** Es un campo importante dentro de PLN, su objetivo es el tratamiento computacional de opiniones, sentimientos y subjetividad en textos. Aplicado en este ámbito, permite clasificar si el tuit es positivo, negativo o neutral. P. ej: Que maravilloso día hace hoy. Es positivo.

Para aplicarlo en este proyecto, se ha utilizado la herramienta Pysentimiento³ ya que incorpora algoritmos de aprendizaje automático y ofrece soporte para español e inglés. Esta herramienta incorpora su propio preprocesador de texto, por lo tanto para usar esta herramienta, obviamente el mejor preprocesador va a ser el que el propio desarrollador ha proporcionado. PySentimiento ofrece tres datos de cada tuit, con los que obtendremos datos de análisis de sentimientos, emociones y discurso de odio.

3. **Análisis de emociones:** Esta parte de PLN va más allá del análisis de opinión, es decir que no solo clasifica el tweet en positivo, negativo o neutro, sino que permite la extracción automática de emociones (ira, amor, vergüenza...) a partir de información textual o hablada. Las emociones que PySentimiento es capaz de detectar son: Alegría, sorpresa, disgusto, tristeza, miedo y enfado, también devuelve el porcentaje de emociones que no ha podido clasificar marcándolo como “otras”.

3 <https://github.com/pysentimiento/pysentimiento>

4. Análisis del discurso de odio [22] [23]: Supone un gran desafío para el PLN, ya que el determinar si un tweet contiene odio o no, puede ser un tema muy subjetivo, además, la búsqueda de corpus, datasets, y modelos de evaluación puede ser compleja. Según las políticas de uso de Twitter, se considera odio:

“Conductas de incitación al odio: no se permite fomentar la violencia contra otras personas ni atacarlas o amenazarlas directamente por motivo de su raza, origen étnico, nacionalidad, pertenencia a una casta, orientación sexual, género, identidad de género, afiliación religiosa, edad, discapacidad o enfermedad grave. Tampoco permitimos la existencia de cuentas cuyo objetivo principal sea incitar la violencia contra otras personas en función de las categorías antes mencionadas.”

3 ANÁLISIS

3.1 Especificación de Requisitos

3.1.1 Requisitos funcionales

Son aquellos que van a determinar lo que hace el sistema o lo que se espera que haga, es decir, su funcionalidad [18]. Para este proyecto los requisitos funcionales identificados son:

- El usuario podrá recuperar tweets en tiempo real o del pasado.
- El sistema debe permitir que las búsquedas se realicen mediante localización geográfica o atendiendo a palabras clave.
- El usuario tendrá la posibilidad de recuperar tweets en español e inglés.
- El sistema debe ser capaz de analizar si una opinión es positiva o negativa, la emoción que expresa el usuario con su comentario y si este contiene lenguaje ofensivo.
- El usuario podrá visualizar de qué dispositivos se publican más tweets, hashtags más repetidos, palabras más frecuentes y usuarios más influyentes.
- El usuario podrá consultar y descargar la información recuperada para poder ser procesada en otro momento u otro software.

3.1.2 Requisitos no funcionales

Los requisitos no funcionales describen aspectos del sistema que están relacionados con el grado de cumplimiento de los requisitos funcionales:

- Clasificación de los mensajes extraídos en tiempo real o pasado.

- Optimización del cómputo de los tweets para ofrecer el mínimo tiempo de respuesta posible.
- Realización de una interfaz web.
- Notificar los errores en tiempo de ejecución que se puedan producir al usuario.
- Posibilidad de almacenar en el sistema los conjuntos de datos creados en tiempo real.

3.2 Planificación Temporal

Para la realización de la planificación temporal se ha utilizado el software GanttProject⁴. El propósito principal de GanttProject es la elaboración de diagramas de Gantt de una forma simplificada, además, con esta aplicación podremos generar diagramas PERT y asignar recursos (materiales y/o humanos) a las diferentes tareas del proyecto, por lo que obtendremos qué recursos y cuántos de ellos necesitaremos a lo largo de su desarrollo.

3.2.1 Diagrama de Gantt

El diagrama de Gantt es una técnica utilizada para la representación gráfica de la estimación temporal de las duraciones de las distintas tareas que componen un proyecto. Esta técnica también nos permite relacionar tareas entre sí, existen cuatro tipos de relaciones:

- Fin-Inicio: Cuando acaba la tarea A, la tarea B empieza.
- Inicio-Fin: La tarea A no puede empezar hasta que acabe B.
- Fin-Fin: La tarea A y B deben comenzar al mismo tiempo.
- Inicio-Inicio: La tarea A y B deben terminar al mismo tiempo.

En el diagrama de Gantt desarrollado para este proyecto (Ilustración 3.1), se comienza con las tareas de “Documentación” y “Análisis de Requisitos”, la primera va a estar presente durante todo el ciclo de vida del proyecto. Una vez obtenidos los requisitos necesarios, siguen las tareas de “Elección del diseño web” y “Creación del proyecto” que consisten en que una vez conocidos los requisitos, elegir qué diseño y arquitectura de la aplicación se adapta mejor.

Cuando se tenga un proyecto y un diseño sobre el que trabajar, las tareas de implementación se desarrollan de forma paralela según las necesidades que vayan surgiendo, por ejemplo: Se desarrolla la funcionalidad para recuperar tweets, se desarrolla el

4 <https://www.ganttproject.biz/>

método para generar estadísticas a partir de los tweets recuperados y se realiza la interfaz para mostrar los resultados. Este proceso se repite para cada caso de uso.

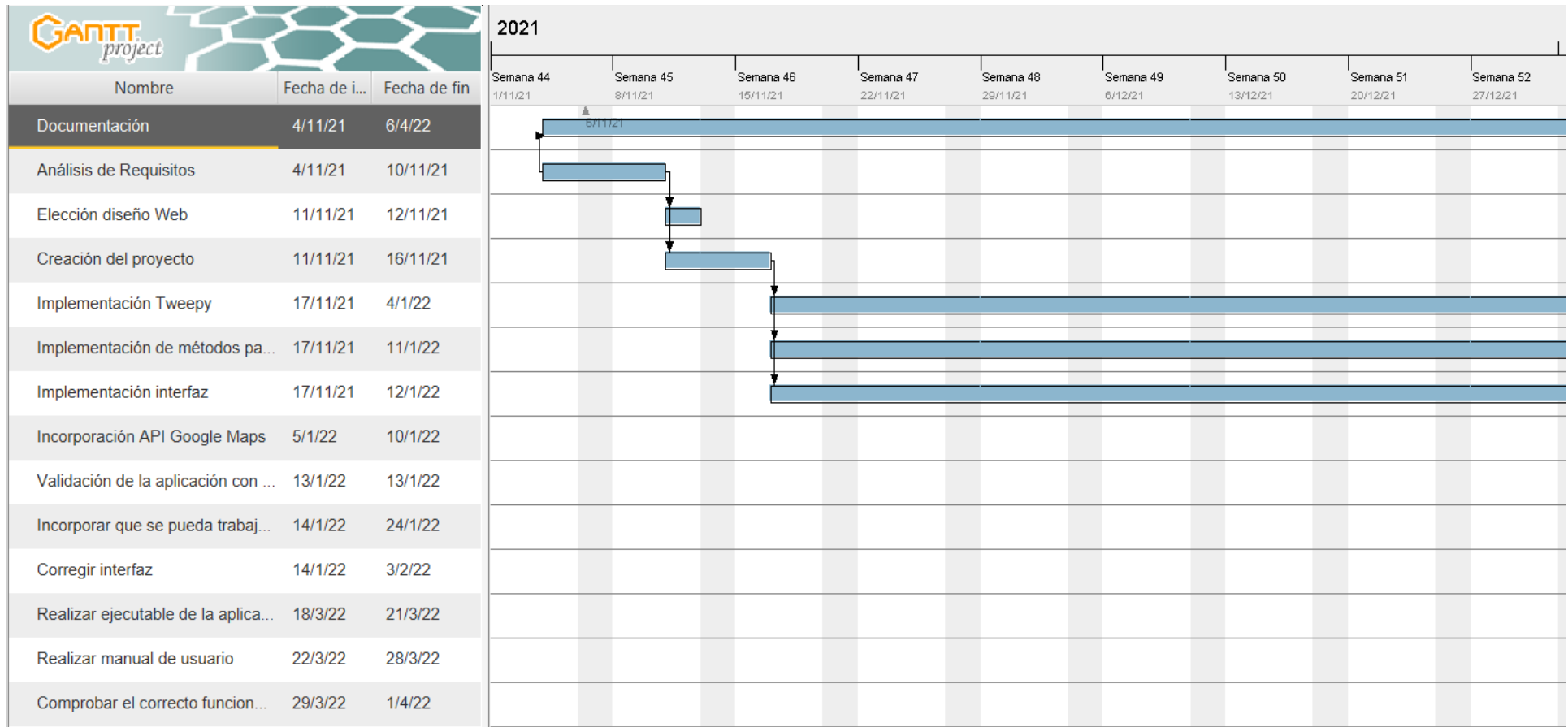


Ilustración 3.1. Diagrama de Gantt I

Cuando se tiene la implementación prácticamente acabada (Ilustración 3.2) se puede incluir la API de Google Maps, ya que esta API en sí, no añade ninguna funcionalidad a la aplicación, solo se usa como interfaz para seleccionar o mostrar ubicaciones, por lo que al tratarse de una API de pago, es preferible usarla en las fases finales del proyecto para así aprovechar al máximo la prueba gratuita que ofrece Google de la misma.

En este punto ya se tiene una versión estable que debería de estar presentable para una revisión completa de la aplicación por parte de los tutores, en esta reunión se propusieron una serie de modificaciones que supusieron un mayor esfuerzo de implementación.

Una vez acabada toda la implementación, se realiza un ejecutable y un manual para el usuario, así como una comprobación del correcto funcionamiento de la aplicación, todo esto documentándose hasta el final.

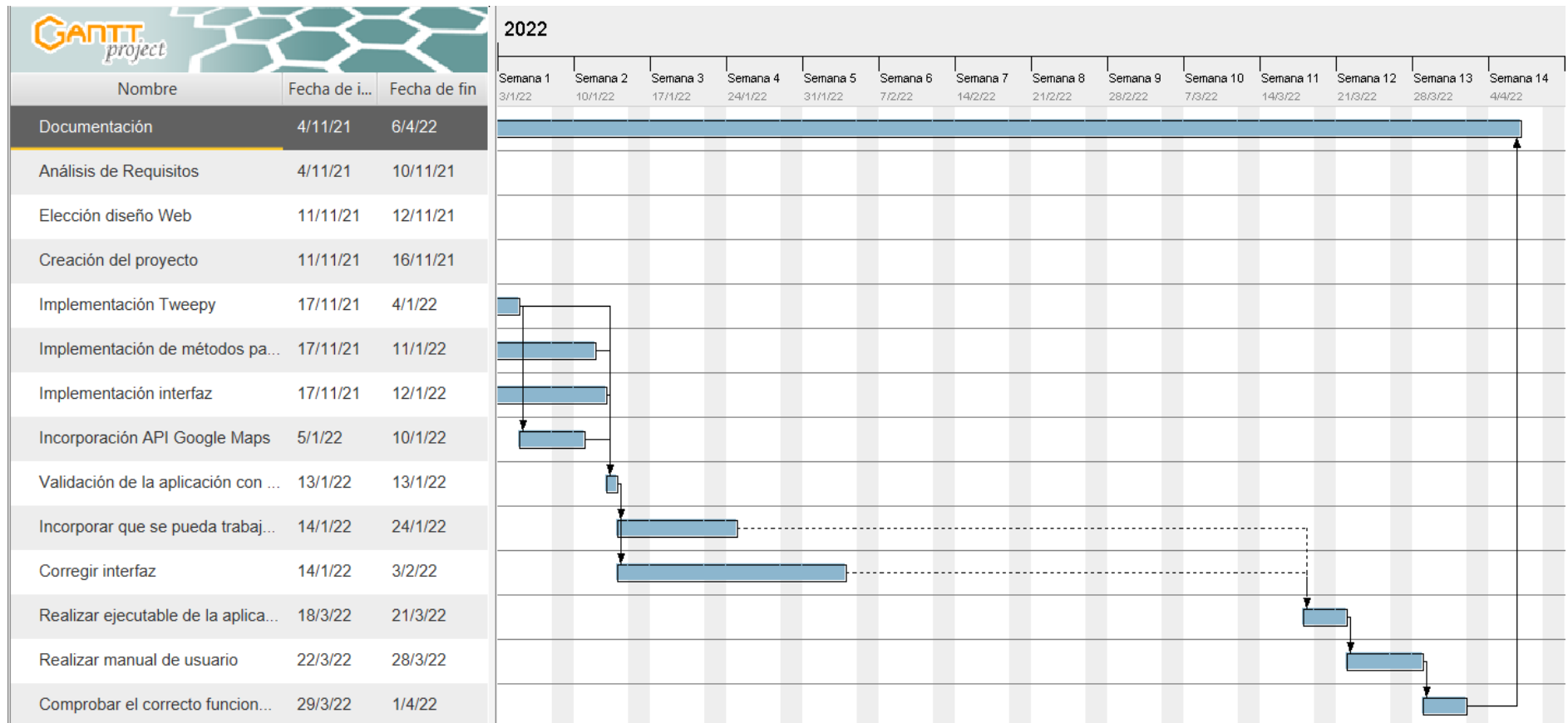


Ilustración 3.2. Diagrama de Gantt II

3.2.2 Diagrama PERT

La aplicación “GanttProject” usada para la elaboración del diagrama Gantt, permite generar un diagrama PERT a partir del de Gantt, este diagrama (Ilustración 3.3), a diferencia del de Gantt, permite establecer el camino a seguir para que la fecha de finalización del proyecto se altere lo mínimo posible.

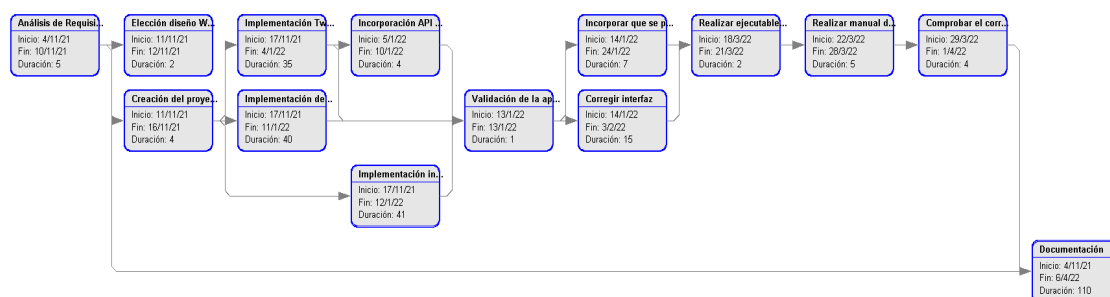


Ilustración 3.3. Diagrama PERT

3.3 Análisis de costes

3.3.1 Costes de desarrollo

Para el desarrollo se ha empleado software libre y gratuito, por lo que el costo de desarrollo viene dado solamente por los gastos en recursos humanos, suministro eléctrico y de Internet. Se ha considerado que en el proyecto intervienen un analista y un desarrollador web. Para estimar sus salarios se ha usado indeed, que es una web que ofrece ofertas de trabajo y además incorporar una herramienta para buscar sueldos⁵:

- El salario promedio de programador/a web en España es de 24 375 € por año. (2031.25€ mensuales).
- El salario promedio de analista programador/a en España es de 27 856 € por año. (2321.33€ diarios).

3.3.2 Costes de funcionamiento

Los costes de funcionamiento varían en función de las peticiones que vaya a recibir el servidor, por ello se exponen las distintas tarifas de las APIs utilizadas y servicios de hosting que se han elegido para el cálculo de costes del funcionamiento de la aplicación.

- Se ha optado por Google Cloud Hosting, como servicio de almacenamiento y ejecución del servidor web. Se ha optado por un plan con optimización para el cómputo el cual contiene:

5 <https://es.indeed.com/career/salaries?from=gnav-homepage>

- La instancia se ejecutaría en Londres, localización más cercana a España.
- Incorpora sistema operativo Linux, 16 vCPUs, 16 GB de RAM y 750 GB de almacenamiento SSD.
- El precio sería de 0.668 USD por hora bajo demanda.

La aplicación usa la API de Google Maps⁶ que ofrece 200 USD/mensuales de uso gratuito, lo que equivale a 28500 solicitudes mensuales, más que suficiente para un uso habitual de la aplicación, si se excediese, se cobraría 2 USD mensuales por cada 1000 peticiones extra.

El uso actual que se hace de la API de Twitter es gratuito, el plan gratuito ofrece:

- 100 Tweets por solicitud.
- Consultas de de hasta 256 caracteres.
- 30 solicitudes por minuto.
- Máximo 25 000 tweets para recuperar para recuperar.

Si se necesitase más, se podría contratar un plan de pago. El límite que tenga la API es independiente a la aplicación, es decir, podemos cambiar el plan sin tocar una sola línea de código de la aplicación. Las tarifas disponibles son (Ilustración 3.4):

	Total Requests PER MONTH ⓘ	Month-to-month PRICE PER MONTH ⓘ
Paid		
	Up to 500	\$149.00
	Up to 1000	\$289.00
	Up to 2,500	\$699.00
	Up to 5,000	\$1,299.00
	Up to 10,000	\$2,499.00
Looking for more? Our Enterprise packages include advanced features and APIs to take full advantage of the opportunities within Twitter data. They also include email support and a dedicated account manager.		
Contact us for enterprise access		

Ilustración 3.4. Planes Tweepy

⁶ https://mapsplatform.google.com/intl/es_ALL/pricing/

3.3.3 Cálculo de costes

El primer paso es identificar los recursos empleados en el desarrollo para posteriormente adjudicar cada recurso al gasto que conlleva. Para la electricidad he tenido en cuenta que un ordenador gasta 10.80 €/mensuales⁷ con un uso de 8 horas diarias y como hay dos ordenadores, uno para cada trabajador, se cuenta 21.60€/mensuales. Para el gasto en la conexión a Internet se ha contratado una tarifa⁸ que supone 34.92 €/mensuales.

Cálculo:

1. Fecha estimada de Fin Proyecto (04/06/2022) – Fecha de Inicio Proyecto (04/11/2021) = 7 meses.
2. Sueldo mensual desarrollador (2031.25€) + Sueldo mensual analista (2321.33€) + Precio mensual internet (10.80€) + Precio mensual electricidad (21.60€) = 4384.98 € de gastos mensuales de desarrollo.
3. Gastos mensuales de desarrollo (4384.98€) x Número de meses de desarrollo (7) = 30 694,86€.

Para los gastos de funcionamiento se ha optado por el Hosting de Google Cloud, que dadas sus características, es muy parecido al de AWS, pero es bastante más barato. Los precios van a variar según la demanda que se requiera del servicio, no obstante para el cálculo del precio, considero que la instancia se va a estar usando 24 horas al día, para ponernos en el “peor” de los escenarios, esto supondría:

$$0.668 \text{ €/hora} \times 24 \text{ horas} \times 30 \text{ días/mes} = 480.96 \text{ €/mensuales.}$$

7 <https://hardzone.es/tutoriales/mantenimiento/medir-consumo-pc/>

8 <https://www.vodafone.es/c/particulares/es/productos-y-servicios/vodafone-one/alt/e/paquete-one-1/>

4 DISEÑO

4.1 Casos de uso

Los casos de uso exponen las distintas actividades las cuales podrá desarrollar el usuario cuando utilice la aplicación, para cada caso de uso se ha detallado los pasos que debe seguir un usuario para realizar cada tarea, además se han incluido diagramas de caso de uso para ayudar a la comprensión de los mismos, concretamente tenemos un diagrama de caso de uso general (Ilustración 4.1) y un diagrama para cada caso de uso (Ilustración 4.2, 4.3, 4.4 y 4.5).

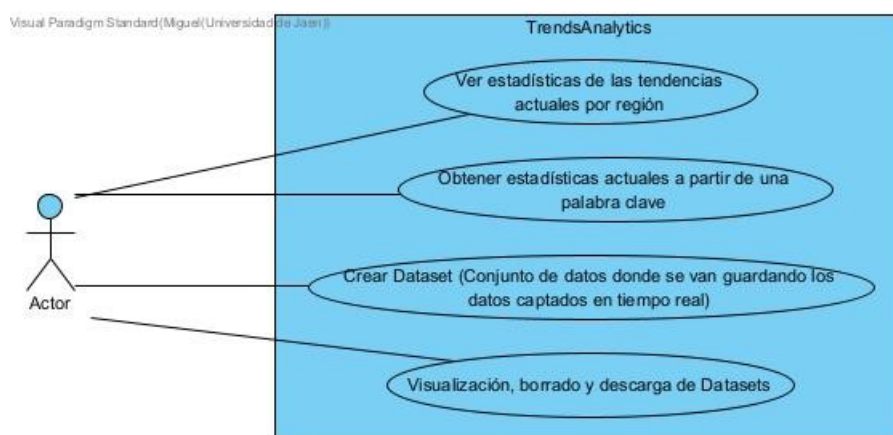


Ilustración 4.1. Diagrama de caso de uso (General)

4.1.1 Ver estadísticas de las tendencias actuales por región

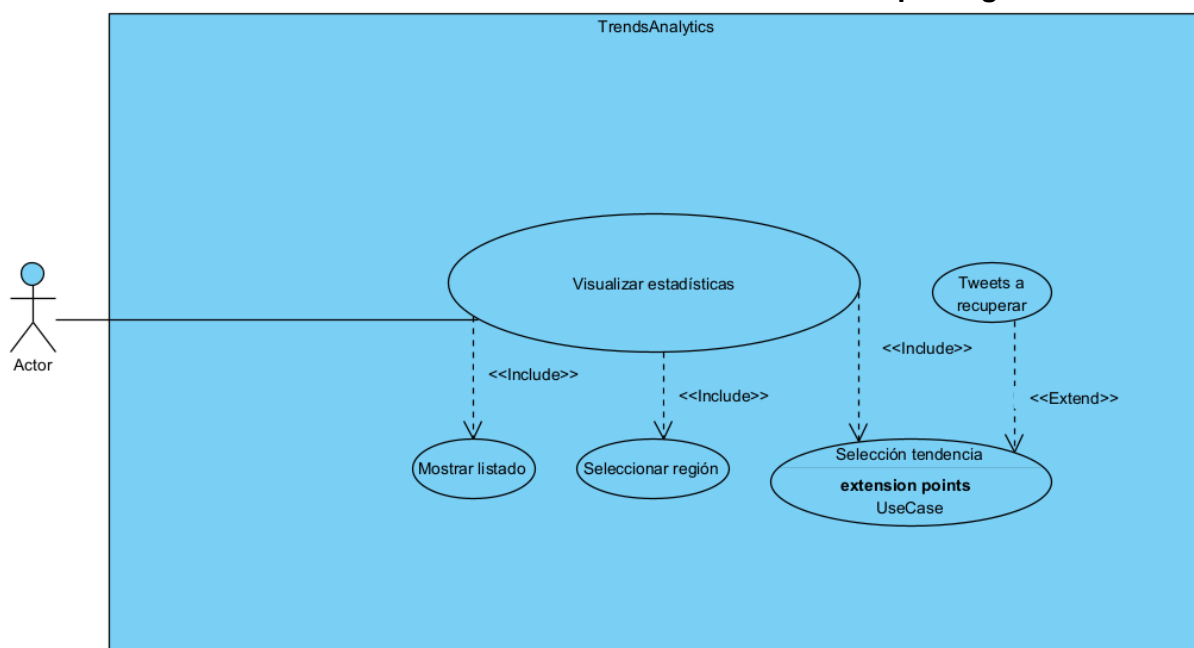


Ilustración 4.2. Diagrama de caso de uso (Ver estadísticas de las tendencias actuales por región)

- I. El usuario abre la lista desplegable de localizaciones disponibles de la barra de navegación, concretamente se encuentra en la esquina superior derecha de la ventana.
- II. Selecciona la región deseada, el número de tweets a recuperar y pulsa sobre el botón de búsqueda.
- III. Se muestra una pantalla con el listado de las tendencias actuales y se pulsa sobre la tendencia de la cual queremos ver las estadísticas.
- IV. Las estadísticas aparecerán en la parte derecha del listado junto con una barra de navegación que nos permite visualizar distintas gráficas.

4.1.2 Obtener estadísticas actuales a partir de una palabra clave

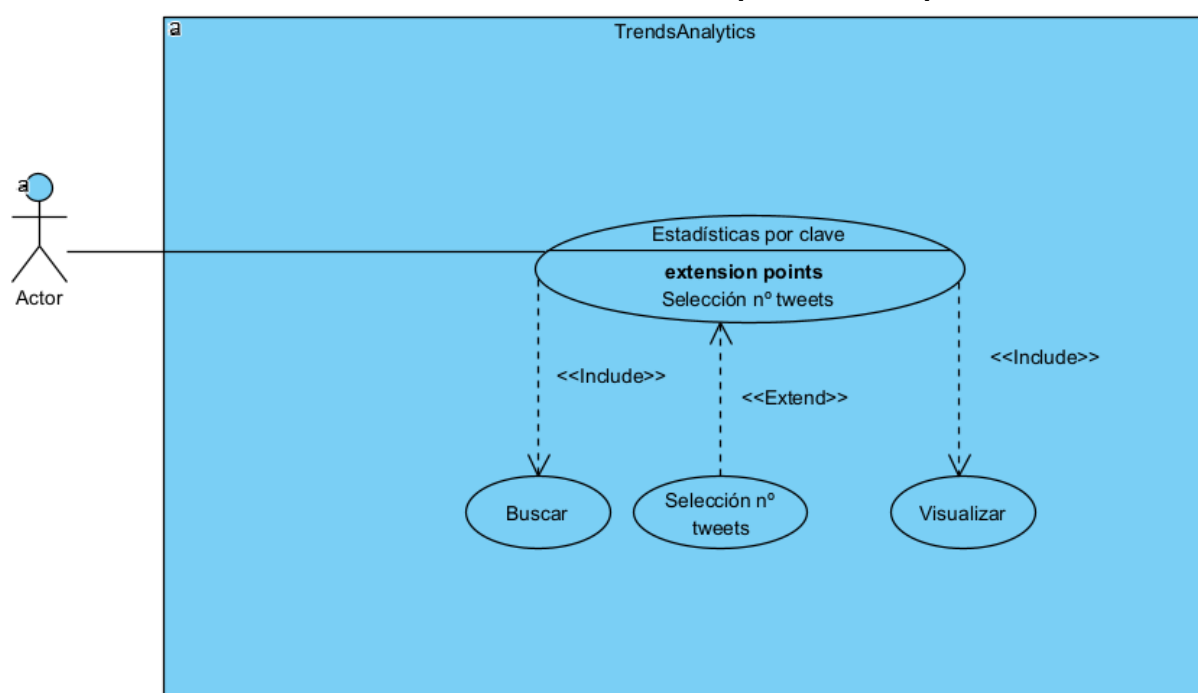


Ilustración 4.3. Diagrama de caso de uso (Obtener estadísticas actuales a partir de una palabra clave)

- I. El usuario pulsa sobre la opción “Search Word” de la barra de navegación.
- II. Se abre una vista donde el usuario puede escribir la palabra clave por la cual hacer la búsqueda, también puede seleccionar el número de tweets a recuperar. A continuación se pulsa sobre el botón de búsqueda.
- III. Las estadísticas aparecerán en la parte inferior del listado junto con una barra de navegación que nos permite visualizar distintas gráficas.

4.1.3 Crear Dataset (Conjunto de datos donde se van guardando los datos captados en tiempo real)

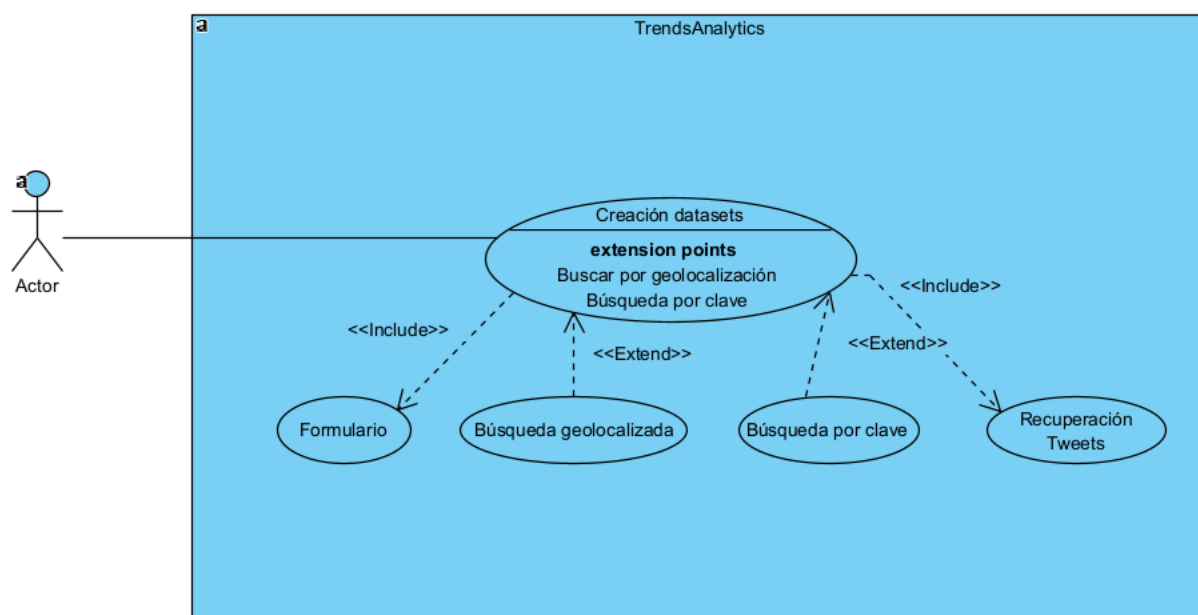
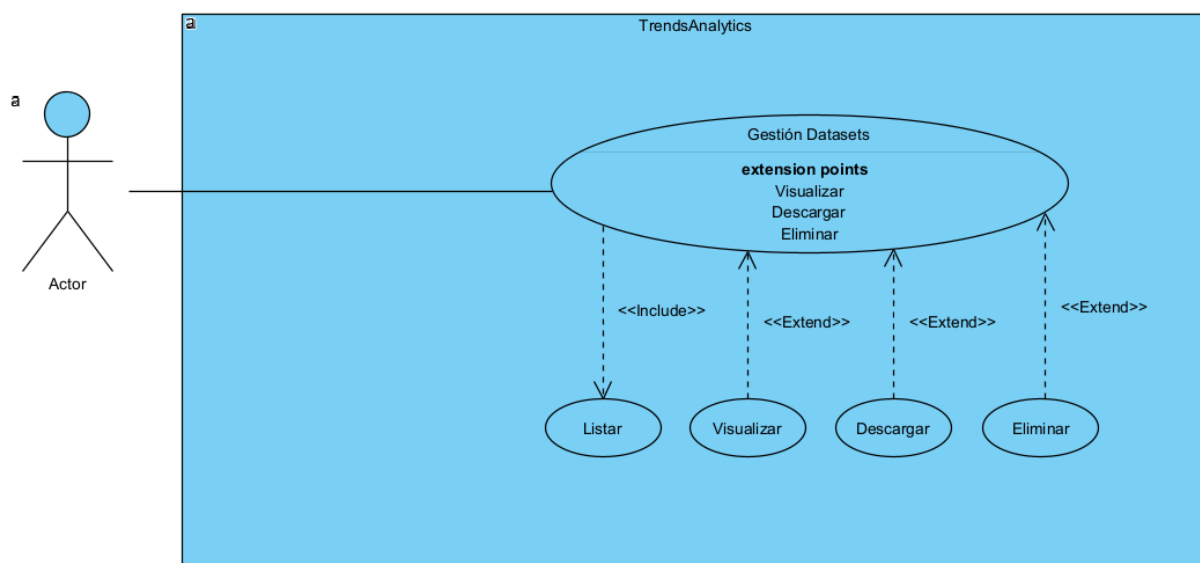


Ilustración 4.4. Diagrama de caso de uso (Crear Dataset)

- I. El usuario debe pulsar el botón “My Datasets” de la barra de navegación. Cuando cargue la vista, pulsar sobre “Create”.
- II. Se abre un formulario el cual el usuario debe completar los siguientes campos:
 - Nombre del conjunto de datos para poder identificarlo fácilmente, este debe ser único.
 - Número máximo de tweets a recuperar.
 - Lenguajes de los que se quiere recuperar tweets.
 - Seleccionar si se quiere recuperar tweets por;
 - Localización: Colocar punto en el mapa.
 - Palabras clave: Escribirlas en el buscador.
- III. Se pulsa sobre el botón crear, desde ese momento se están captando tweets hasta que se llegue al límite asignado.

4.1.4 Visualización, borrado y descarga de Datasets



IV. Ilustración 4.5. Diagrama de caso de uso (Visualización, borrado y descarga de datasets)

- I. El usuario debe pulsar el botón “My Datasets” de la barra de navegación.
- II. Debajo de la barra de navegación, hay una lista desplegable con todos los nombres de los datasets creados. Se selecciona con el que se quiera trabajar.
- III. Una vez seleccionado, el usuario puede borrarlo o descargarlo. También puede visualizar un resumen de sus datos en una tabla, esta tabla contiene los campos:
 - Text: Contenido textual de tweet.
 - Source: Dispositivo o plataforma desde donde se ha publicado el tweet.
 - Ciudad: Ciudad desde la cual se ha publicado el tweet (Si se ha publicado con la localización activada).
 - Language: Idioma en el que está escrito el tweet.
 - Emotion: Indica si el tweet es positivo, neutral o negativo.

4.2 Diagramas de secuencia

En esta sección se exponen los diagramas de secuencia de las funcionalidades más complejas del sistema. Los diagramas de secuencia expresan las distintas interacciones entre las partes intervinientes del sistema, en este caso, interviene el usuario, la aplicación y las APIs de Twitter y Google Maps. Debajo de cada diagrama de secuencia se muestra una breve descripción de cada interacción.

4.2.1 Ver estadísticas de las tendencias actuales por región

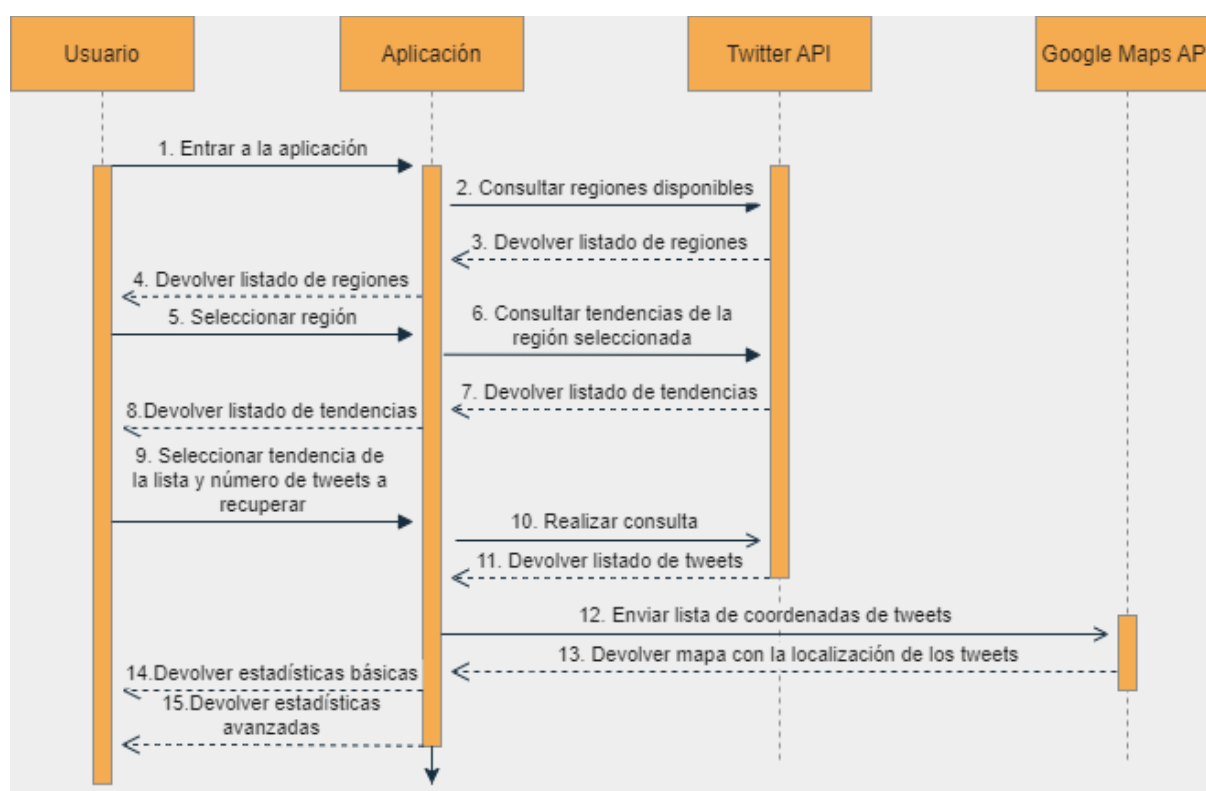


Ilustración 4.6. Diagrama de secuencia (Ver estadísticas de las tendencias actuales por región)

1. Entrar a la aplicación: El usuario entra a la aplicación, el listado de las regiones aparecerá en la barra de navegación, por lo que no importa la ruta a la que acceda.

2, 3 y 4. Consultar regiones disponibles: La aplicación proporcionará un desplegable en el cual el usuario podrá seleccionar. Es la propia API de Twitter la que determina qué regiones están disponibles⁹, para ello usa los identificadores WOEID (Where On Earth Identifier) [19] creados por Yahoo para representar mediante 32 bits localizaciones de forma única. El usuario debe elegir una región del

⁹ <https://codebeautify.org/jsonviewer/f83352>

listado y presionar el botón de búsqueda, para que la aplicación le devuelva el listado de regiones.

5, 6, 7 y 8. El usuario elegirá de la lista una región y la aplicación consultará en la API de Twitter el listado de tendencias actuales de la región seleccionada.

9, 10 y 11. El usuario selecciona una tendencia de la lista y el número de tweets que la aplicación solicitará a la API de twitter, una vez descargados los tweets, se comenzará su cómputo para la generación de estadísticas.

12, 13, 14 y 15. La aplicación ofrece un mapa de geolocalización de los tweets, para ello, obteniendo sus coordenadas, las envía a la API de Google Maps, y esta devuelve el mapa con marcadores colocados en las coordenadas especificadas. Las estadísticas se han agrupado en estadísticas básicas y estadísticas avanzadas, esto se ha diseñado así porque las estadísticas avanzadas son aquellas que se generan a partir de PySentimiento, tardan más en generarse y por eso se envían antes las básicas para que el usuario las pueda ir consultando mientras las avanzadas terminan de generarse.

4.2.2 Obtener estadísticas actuales a partir de una palabra clave

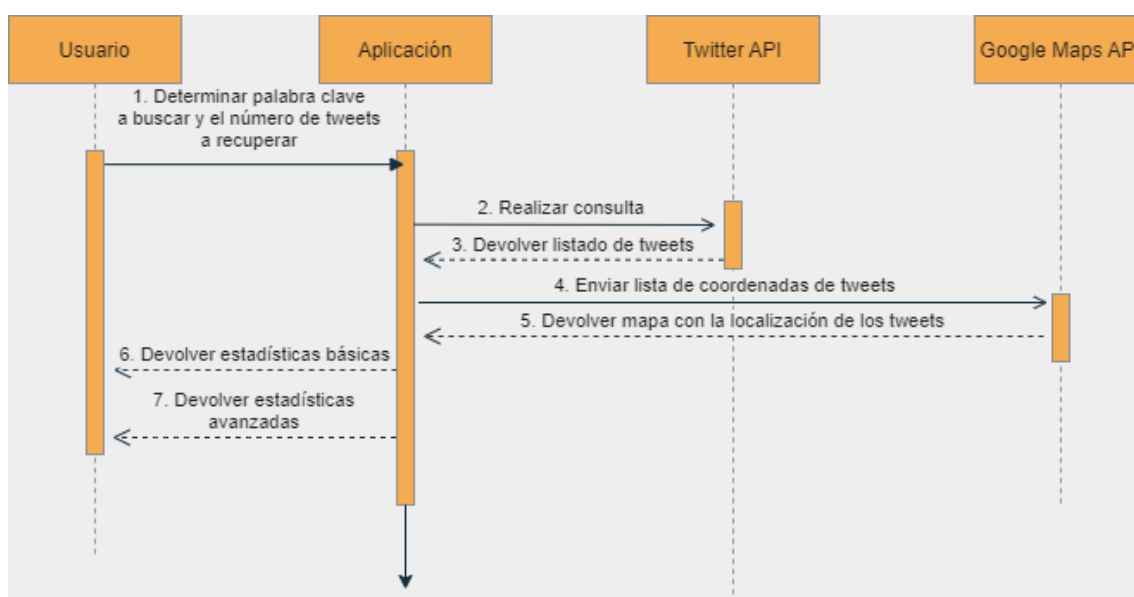


Ilustración 4.7. Diagrama de secuencia (Obtener estadísticas actuales a partir de una palabra clave)

1, 2 y 3. Tanto el diseño, como su implementación deberá de ser idéntico al diagrama de secuencia anterior, con la única diferencia de que la palabra clave no vendrá dada por un listado de tendencias, sino que la proporcionará directamente el usuario junto con el número de tweets a recuperar.

4, 5, 6, y 7. La aplicación ofrece un mapa de geolocalización de los tweets, para ello, obteniendo sus coordenadas, las envía a la API de Google Maps, y esta devuelve el mapa con marcadores colocados en las coordenadas especificadas. Las estadísticas se han agrupado en estadísticas básicas y

estadísticas avanzadas, esto se ha diseñado así porque las estadísticas avanzadas son aquellas que se generan a partir de PySentimiento, tardan más en generarse y por eso se envían antes las básicas para que el usuario las pueda ir consultando mientras las avanzadas terminan de generarse.

4.2.3 Creación de un Dataset y visualización de sus estadísticas

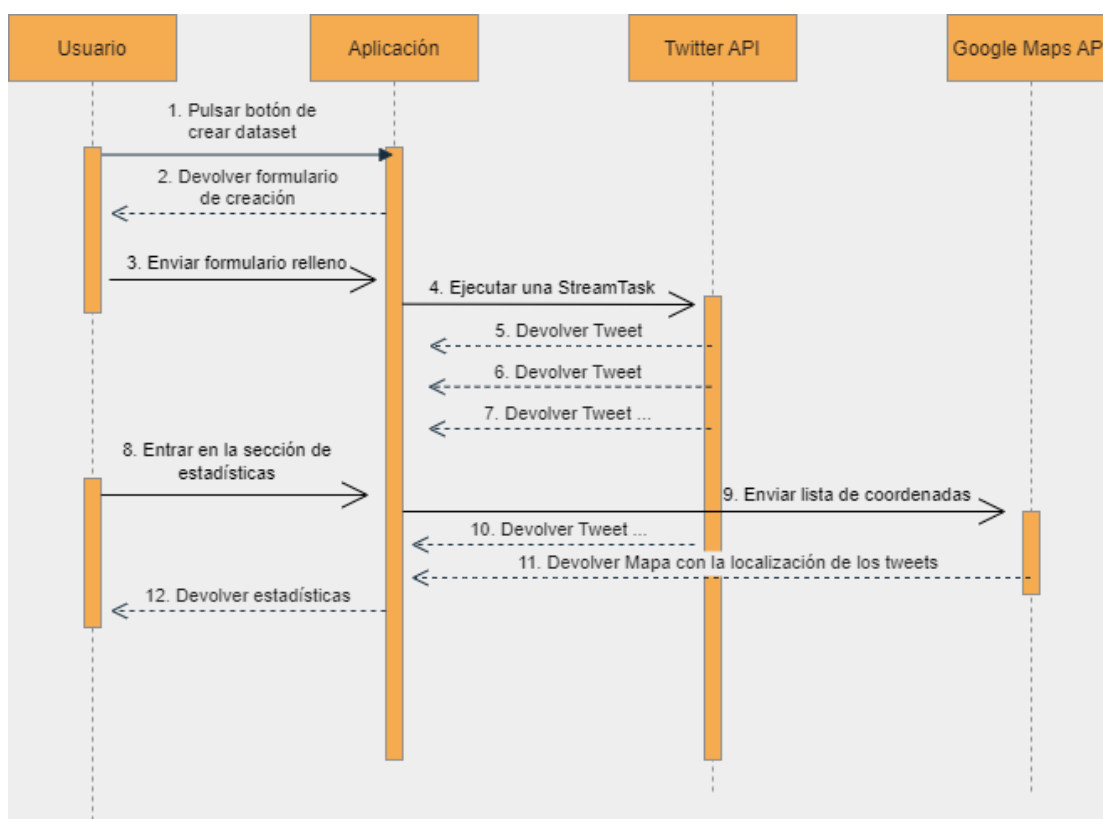


Ilustración 4.8. Diagrama de secuencia (Creación de un Dataset y visualización de sus estadísticas)

1. El usuario deberá acceder a la sección “My Datasets” ubicada en la barra de navegación, en esta vista aparecerá el listado de datasets creados, en la parte superior derecha, el usuario debe seleccionar la opción “Create Dataset”.

2. La vista de creación del dataset está comprendida por un formulario el cual el usuario deberá completar y determinar los siguientes parámetros:

- Nombre del dataset.
- Idiomas en los que recuperar tweets.
- Número de tweets a recuperar.
- Seleccionar si se quiere buscar por palabras clave, e indicar que palabra/s clave se quiere usar, o si se quiere buscar por geolocalización, en este caso, indicar en el mapa la ubicación de la que se requieren tweets.

3 y 4. Una vez el formulario se haya enviado, se crea un StreamTask, es decir, una tarea que va a estar en ejecución hasta que el límite de tweets fijado se alcance. Esta ejecución se realizará en paralelo al resto, por lo que se podrá usar el resto de la aplicación sin problema.

5, 6 y 7. A medida que se vayan recuperando los tweets, se analizarán con PySentimiento de tal forma que cuando el usuario desee obtener las estadísticas, este proceso, que es el más lento, esté ya hecho. Los tweets recuperados, metadatos y estadísticas generadas con PySentimiento, se almacenarán en un archivo en CSV.

8. El usuario en cualquier momento, puede dirigirse a la sección de “My Datasets”, y cuando haya seleccionado el nombre del dataset deseado en el listado de datasets creados, las estadísticas se visualizarán.

9. Cuando se proporcionan las estadísticas, a diferencia de los casos anteriores, se proporcionan en una sola comunicación, ya que, como se ha mencionado, el cómputo más lento ya se ha realizado previamente.

10, 11 y 12. La visualización de las estadísticas se puede realizar mientras se siguen recopilando tweets.

5 PROPUESTA DE SOLUCIÓN

Una vez determinado el análisis del sistema y la aproximación del diseño, en este apartado se detalla el diseño de la arquitectura software y también el diseño de la interfaz web.

5.1 Arquitectura

La arquitectura que se ha elegido se ha dividido en dos diagramas, un diagrama para Frontend y otro para Backend.

5.1.1 Arquitectura Frontend

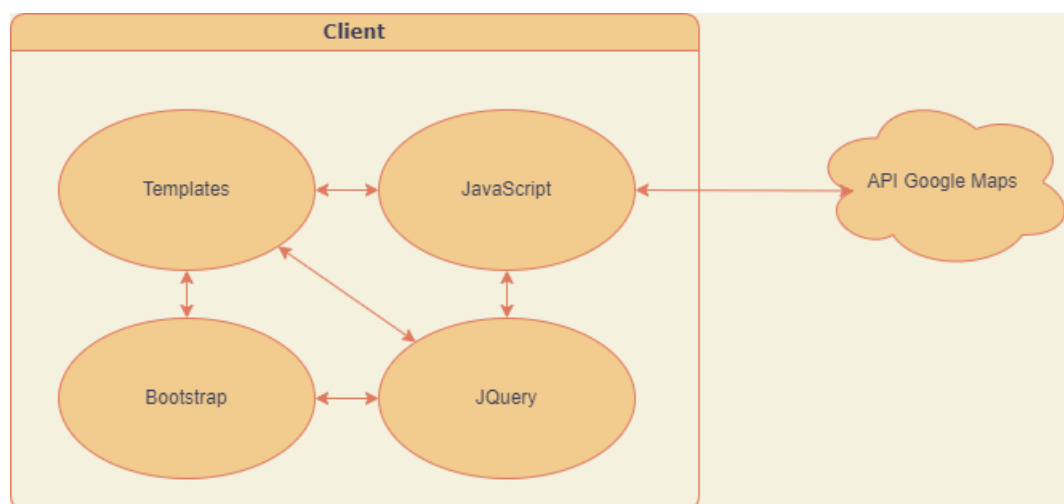


Ilustración 5.1. Diagrama diseño Frontend

La implementación del frontend ha tomado como punto de partida una plantilla de Bootstrap¹⁰ gratuita y disponible en internet. Ha sido necesario incorporar JQuery¹¹ para el uso de plugins de Bootstrap. El uso de Bootstrap proporciona un conjunto de clases predefinidas que dan una gran facilidad de implementación de aplicaciones web responsivas y visualmente atractivas sin necesidad de un gran esfuerzo de implementación de clases propias de CSS.

¹⁰ <https://getbootstrap.com/>

¹¹ <https://jquery.com/>

5.1.2 Arquitectura Backend

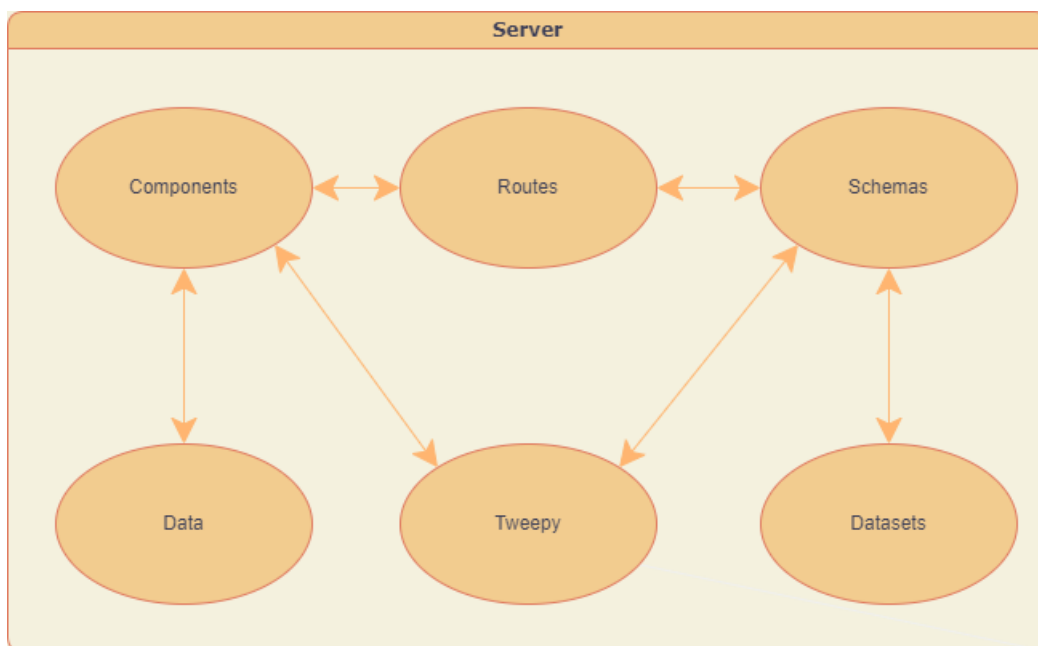


Ilustración 5.2. Diagrama diseño Backend

La arquitectura backend está organizada conforme se indica el diagrama y a continuación se detalla concretamente cómo se organiza:

- Data.
 - Config.cfg: Fichero de configuración de la aplicación que nos permite su personalización a nivel de código.
 - Credentials.json: Fichero que contiene las claves secretas de acceso a las APIs.
- Components.
 - Utils.py: Contiene funciones auxiliares que se van a utilizar en diferentes puntos de la aplicación y proporciona un acceso sencillo a los datos almacenados en "Data"
- Routes.
 - __init__.py: Contiene las rutas de los controladores.
 - Implementation.py: Contiene los métodos que compone la lógica del controlador, de tal forma que se separan las rutas de la lógica del controlador por motivos meramente de organización.
 - StatisticsImplementation.py: Idem que el anterior, pero los métodos son específicos para la generación de las estadísticas.

- Schemas:
 - Tweet.py: Es el único modelo que usa la aplicación, ya que la mayoría de datos se transmiten directamente en json sin necesidad de mapearse en ningún modelo.
- Datasets: Directorio que contiene los datasets creados en formato csv.
- Tweepy:
 - APITwitter.py: Está compuesto por los métodos que llaman a la API de Twitter.
 - StreamTask.py: Es la clase que se usa para instanciar búsquedas de tweets en tiempo real, contiene, además de los métodos propios para la captura de tweets en tiempo real, atributos como son el contador de tweets, límite de tweets y nombre del dataset que se está generando.

5.2 Interfaz

Pantalla principal

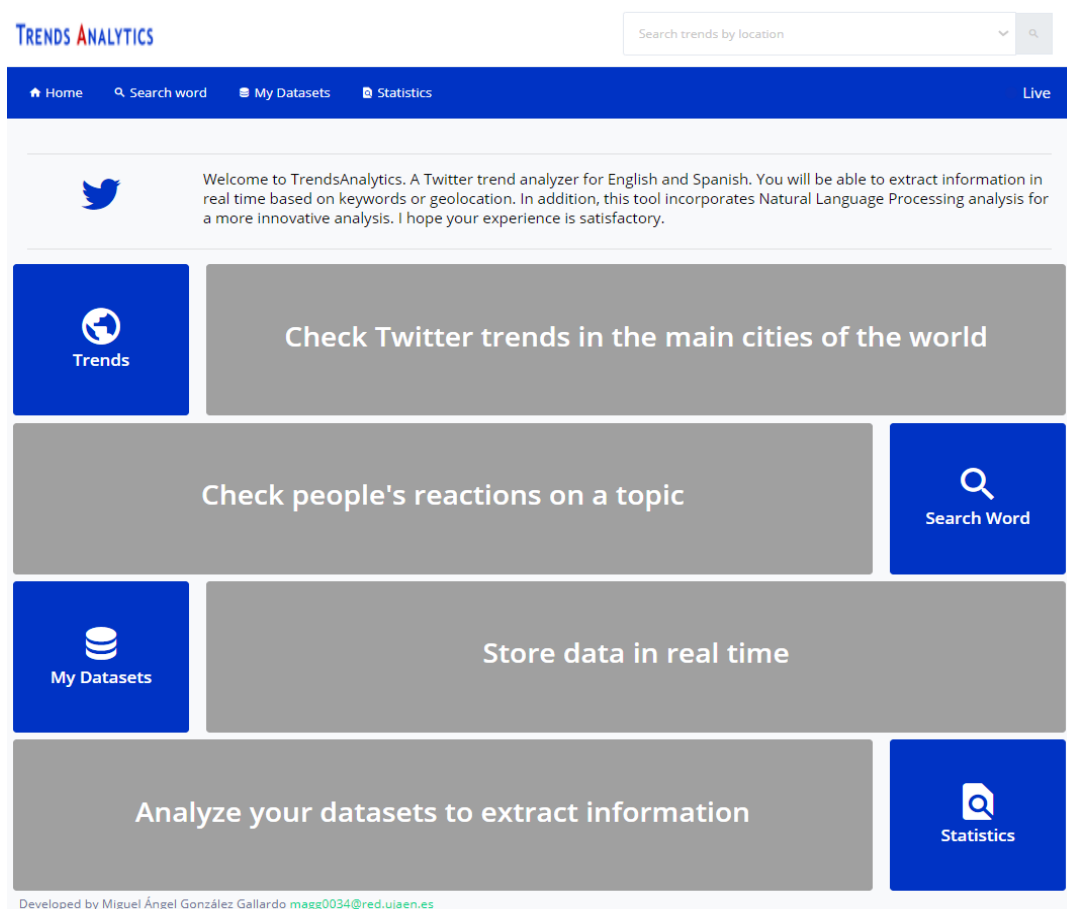


Ilustración 5.3. Interfaz página principal

La página principal (Ilustración 5.3) es la primera pantalla que veremos cuando entremos a la aplicación, en esta vista no tenemos ninguna funcionalidad, simplemente se enumeran brevemente las posibilidades que ofrece la aplicación junto a una breve descripción.

Listado de regiones para obtener sus tendencias

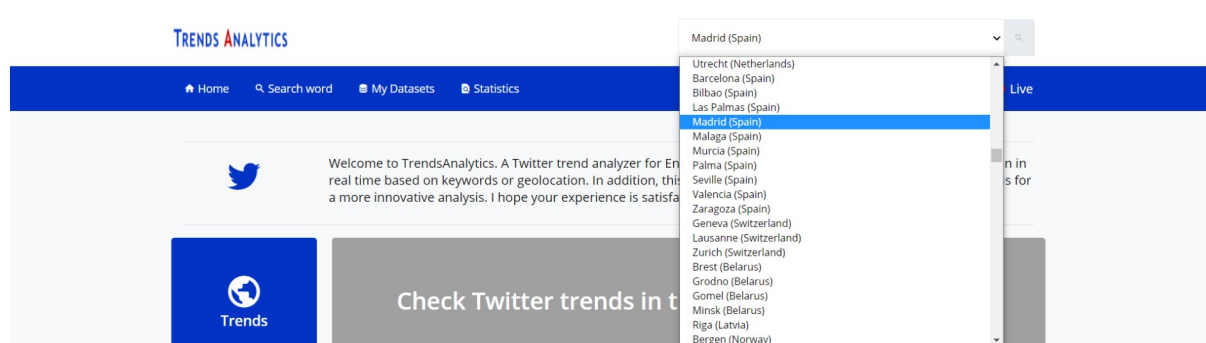


Ilustración 5.4. Interfaz listado de regiones

El listado de regiones (Ilustración 5.4) será accesible desde cualquier vista, ya que se encuentra en la barra de navegación, en ella se listan las regiones que suministra Twitter como puntos de referencia para listar tendencias. Es la misma lista que podemos encontrar en la propia aplicación de Twitter para seleccionar de dónde queremos ver las tendencias.

Listado de tendencias de una región

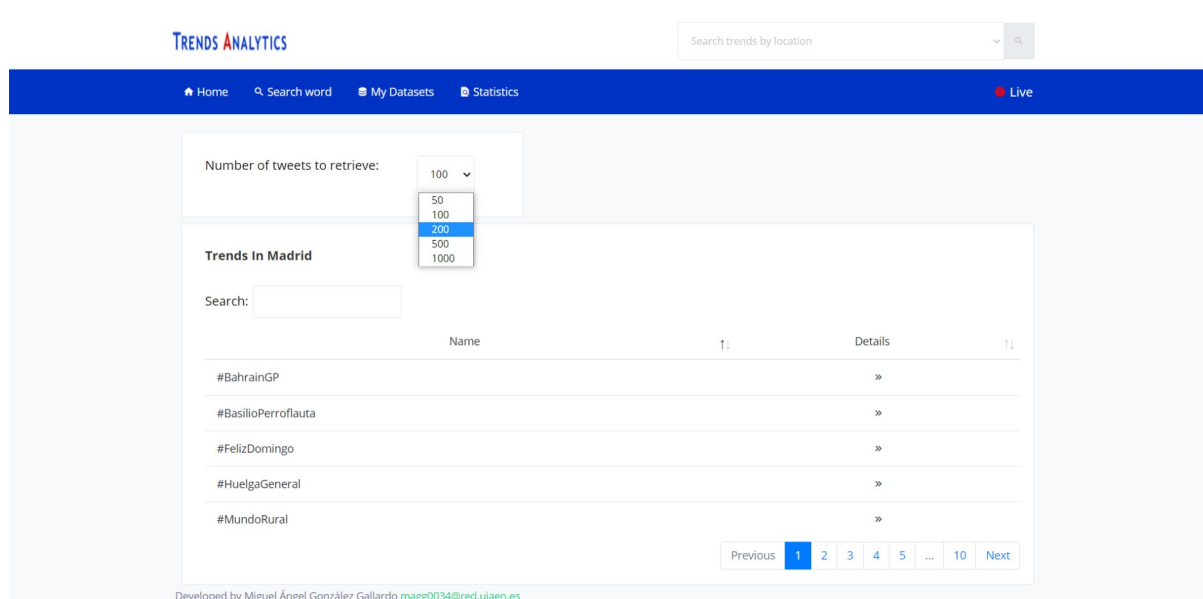


Ilustración 5.5. Listado de tendencias de una región

Una vez seleccionada una región del listado de regiones, se muestra esta vista (Ilustración 5.5) en la cual se muestra el listado de las tendencias actuales de la región seleccionada. Encima del listado podemos seleccionar el número de tweets a recuperar, una vez lo tengamos fijado podemos pulsar sobre la tendencia deseada y en la parte derecha del listado nos aparecerán los tweets recuperados junto a sus estadísticas (Ilustración 5.7).

Búsqueda de tweets por palabra clave

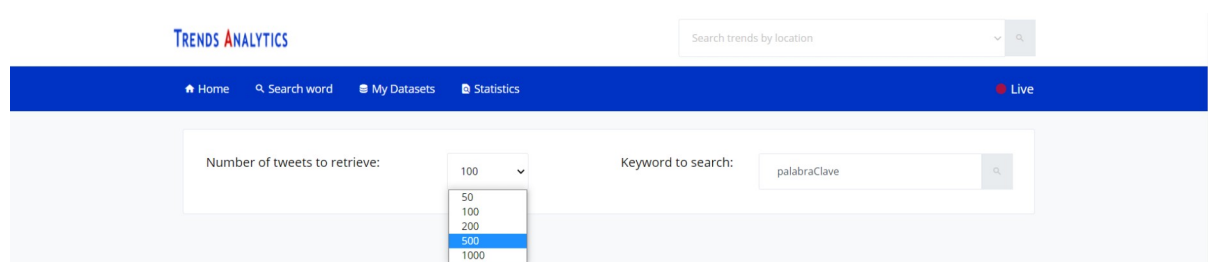


Ilustración 5.6. Interfaz búsqueda de tweets por palabra clave

Esta vista (Ilustración 5.6) se muestra cuando pulsamos la opción “Search word” en la barra de navegación. Su funcionamiento es el mismo que la búsqueda por regiones, con la diferencia que aquí la palabra clave no se obtiene a partir de un listado de tendencias, sino que el usuario puede determinar cualquier palabra clave.

Estadísticas obtenidas a partir de región o palabras clave

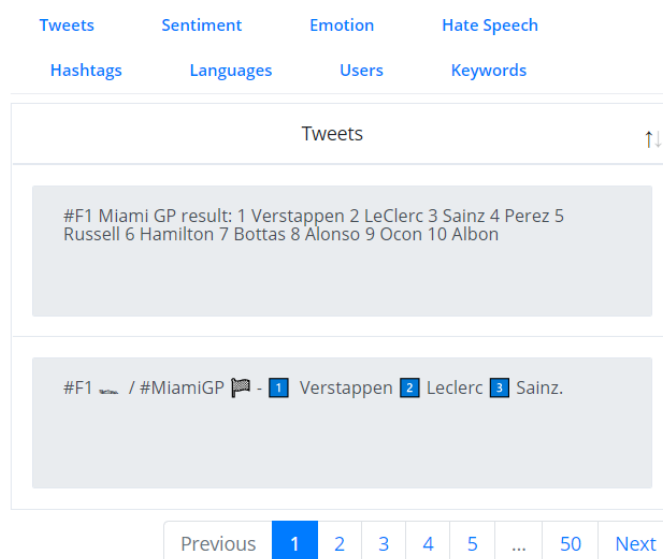


Ilustración 5.7. Interfaz listado de tweets

Este fragmento de vista (Ilustración 5.7) es compartido por las funcionalidades “Buscar por palabra clave” y “Búsqueda por región”, consiste en un listado de tweets de los

cuales se han generado las estadísticas. En la parte superior tenemos pestañas para mostrar las distintas estadísticas generadas, las cuales se muestran a continuación:

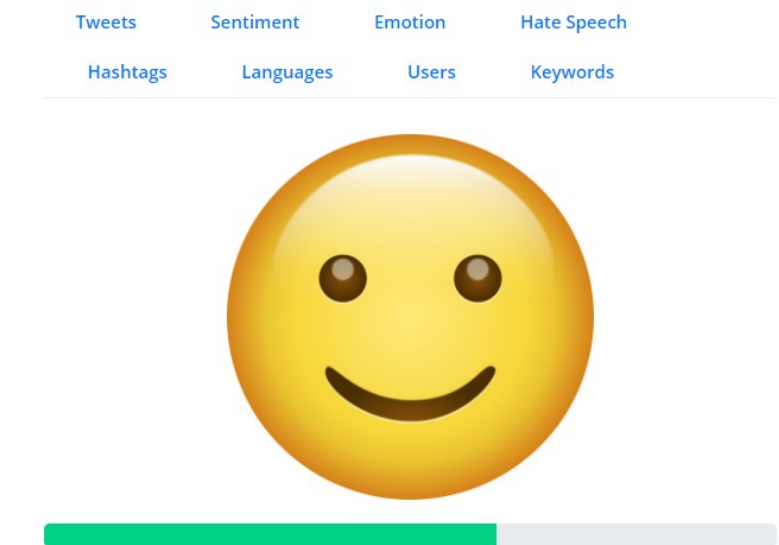


Ilustración 5.8. Interfaz sentimiento

Esta vista (Ilustración 5.8) se muestra en la pestaña “Sentiment”, la cual nos muestra el estado de ánimo que expresan los tweets, a mayor positividad, la cara estará más sonriente, y a más negatividad, tendremos una cara triste, también podemos medir la positividad mediante la barra de progreso que se muestra en la parte de debajo.

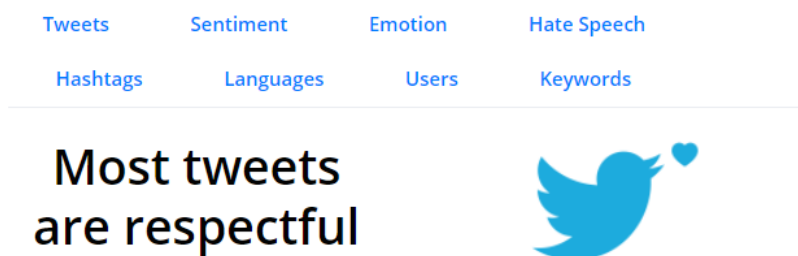


Ilustración 5.9. Interfaz HateSpeech

La pestaña de “Hate Speech” es la encargada de indicarnos si el algoritmo de aprendizaje automático detecta discurso de odio en al menos un cuarto de los tweets recuperados. La razón de incorporar un umbral para indicar la presencia o no de discurso de odio, es que si en un conjunto de tweets, por ejemplo, de 1000 tweets, 20 tweets son detectados como mensajes que incitan al odio, suponen un número tan pequeño que no es relevante para las estadísticas.

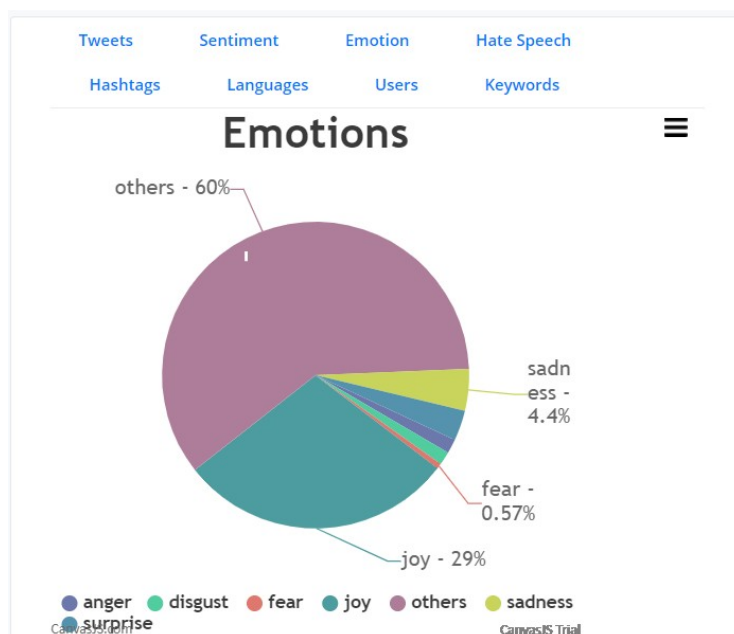


Ilustración 5.10. Interfaz emociones

Esta vista (Ilustración 5.10) se muestra la pestaña de “Emotion”, en su contenido nos aparece una gráfica de sectores con las distintas emociones captadas.

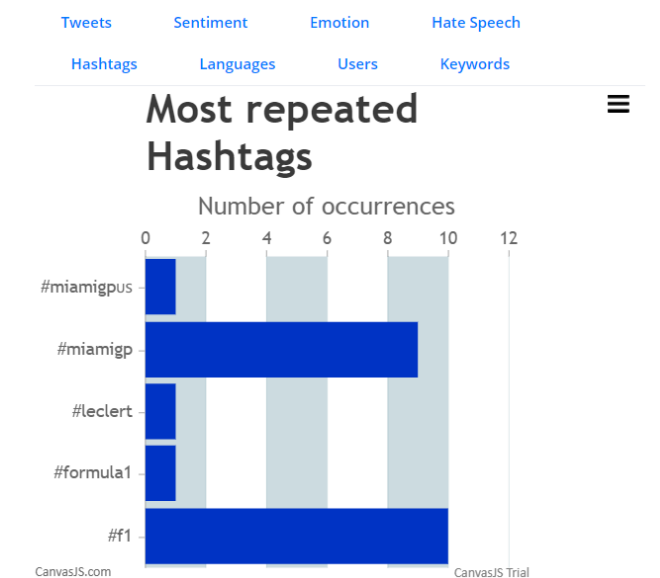


Ilustración 5.11. Interfaz hashtags

En la pestaña “Hashtags” nos aparecen (Ilustración 5.11) los cuatros hashtags más repetidos en el listado de tweets.

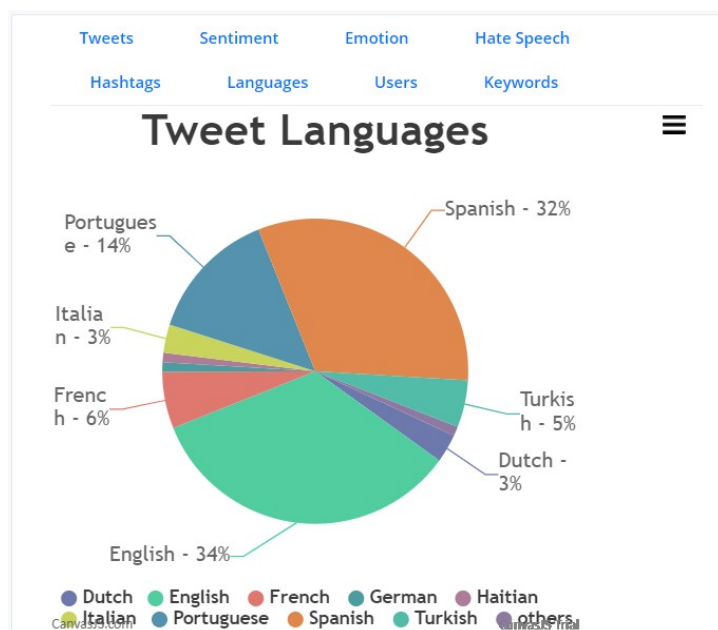


Ilustración 5.12. Interfaz idiomas

En la pestaña “Languages” (Ilustración 5.12) nos aparece un gráfico de sectores con los idiomas de los tweets recuperados.

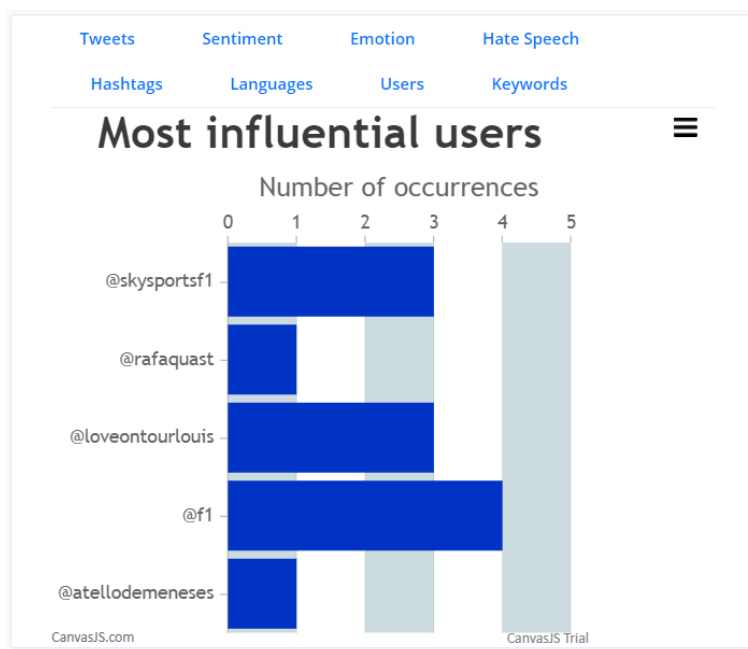


Ilustración 5.13. Interfaz usuarios

En “Users” (Ilustración 5.13) nos aparecen los 5 usuarios más influyentes, es decir, su nombre de usuario aparece más veces o han mandado más tweets en el listado de tweets recuperados.

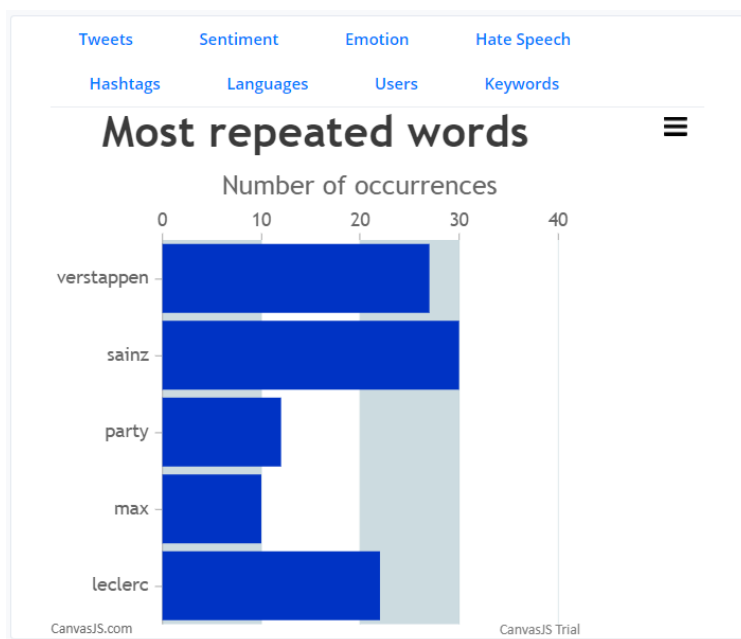


Ilustración 5.14. Interfaz palabras con más ocurrencias

En la pestaña “Keywords” (Ilustración 5.14) nos aparecen las palabras clave, es decir, palabras más repetidas en el listado de tweets.

Vista de administración de Datasets

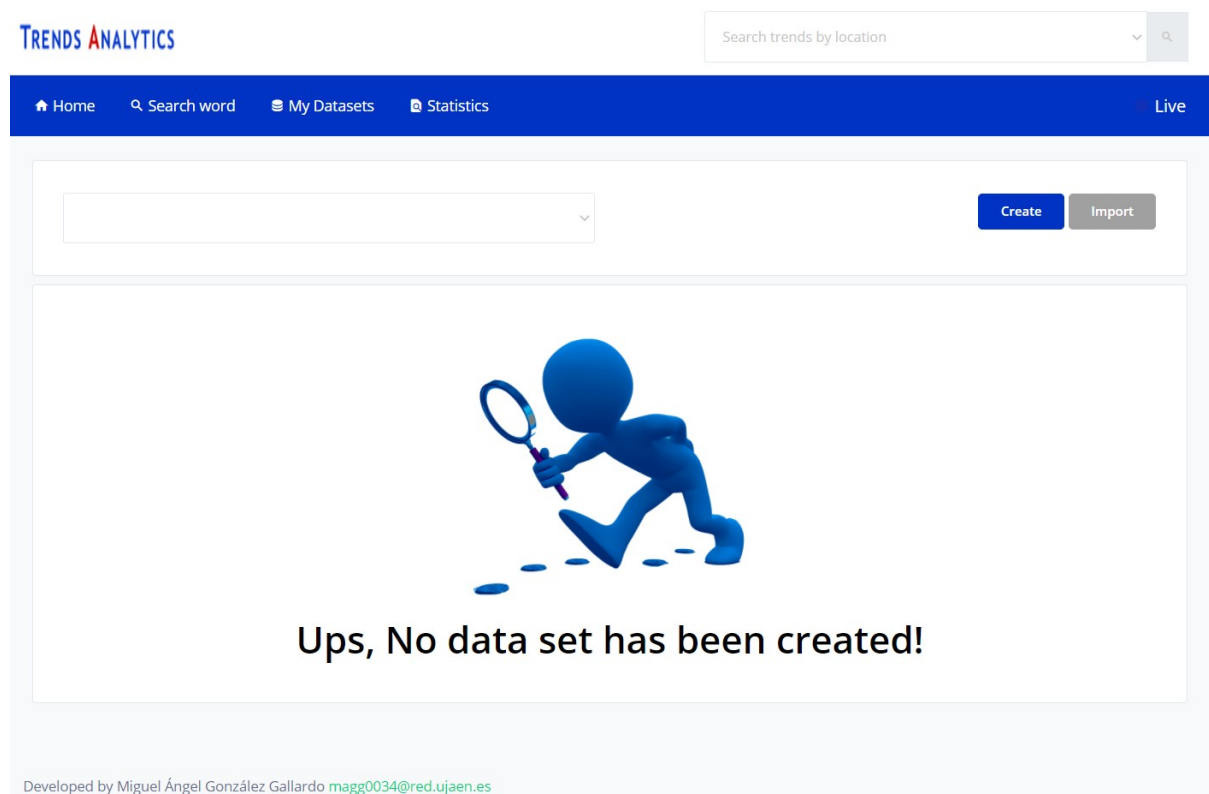
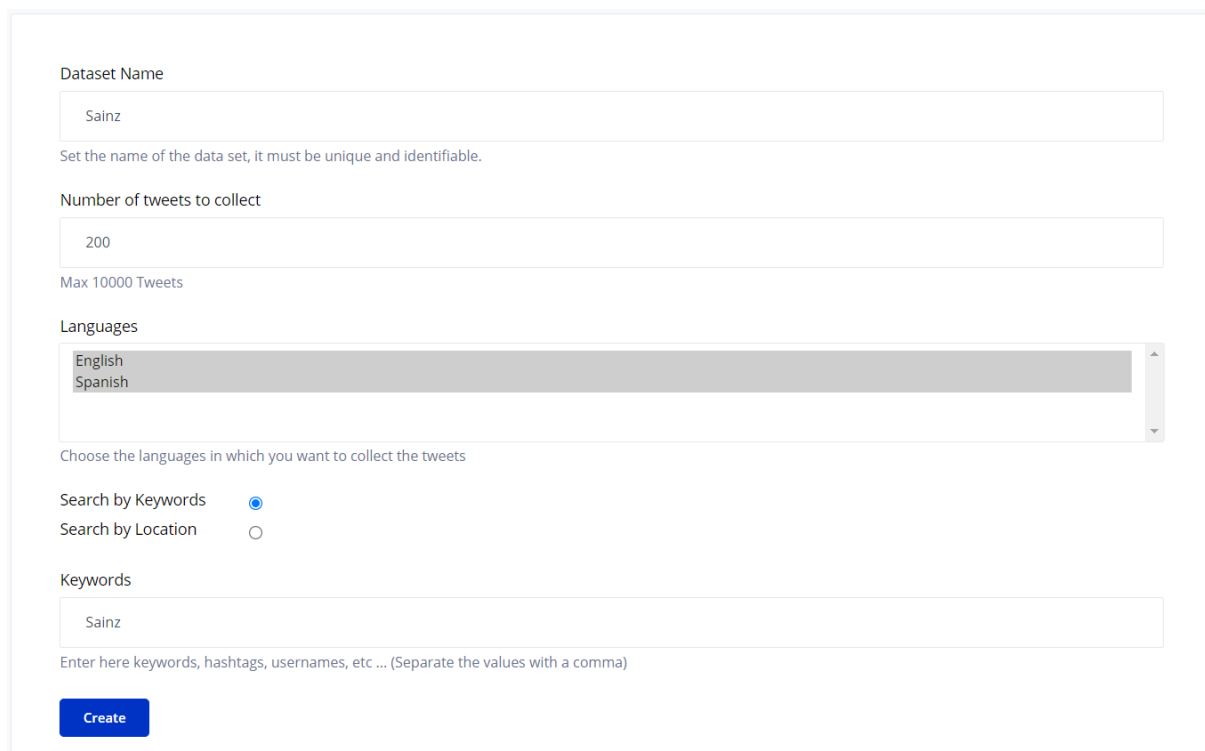


Ilustración 5.15. Interfaz mis datasets creados (vacío)

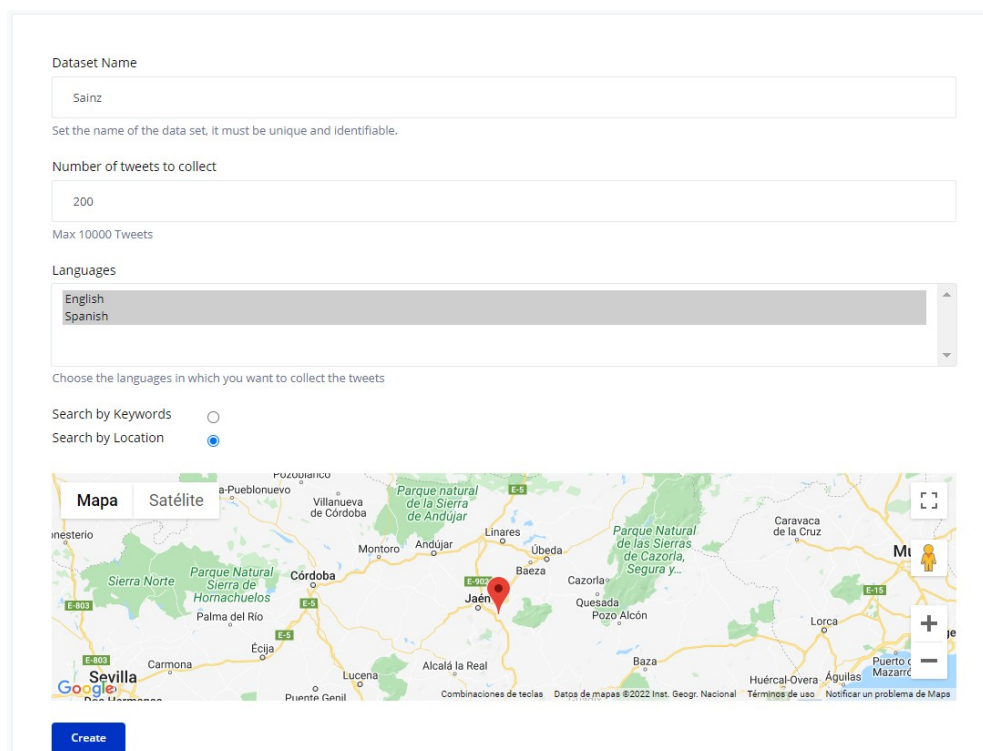
Esta vista (Ilustración 5.15) se mostrará cuando el usuario quiera visualizar los datasets pero no tenga ninguno creado.



The screenshot shows a web form for creating a dataset. It includes a 'Dataset Name' field with the value 'Sainz' and a note: 'Set the name of the data set, it must be unique and identifiable.' Below this is a 'Number of tweets to collect' field with the value '200' and a note: 'Max 10000 Tweets'. The 'Languages' section has a dropdown menu with 'English' and 'Spanish' selected. The 'Search by Keywords' radio button is selected, and the 'Keywords' field contains 'Sainz' with a note: 'Enter here keywords, hashtags, usernames, etc ... (Separate the values with a comma)'. A blue 'Create' button is at the bottom.

Ilustración 5.16. Interfaz formulario de creación de un dataset por palabra clave

Esta vista (Ilustración 5.16) se corresponde al formulario de creación cuando queremos crear un dataset recuperando tweets por palabra clave.



The screenshot shows the same web form as in Illustration 5.16, but with the 'Search by Location' radio button selected. Below the form fields is a Google Map of the Jaén region in Spain, with a red pin indicating the location. The map shows various towns and geographical features. A blue 'Create' button is at the bottom.

Ilustración 5.17. Interfaz formulario de creación de un dataset por geolocalización

Cuando se quiera recuperar tweets por palabras clave, en el formulario se muestra un mapa (Ilustración 5.17) en el cual podemos seleccionar el centro de la región de la cual querámos obtener tweets.

The screenshot shows the Trends Analytics web application. At the top, there's a search bar for trends by location. Below the navigation bar, a dropdown menu shows 'Baloncesto.csv'. To the right are buttons for 'Create', 'Delete', 'Export', and 'Import'. The main area displays a list of tweets with columns for Text, Source, City, Language, and Sentiment. The tweets are sorted by source, showing five entries from 'Twitter for Android' and 'TweetDeck'. The sentiment analysis is shown as a colored circle (blue for neutral, green for positive, red for negative). At the bottom, it indicates 'Showing 6 to 10 of 201 entries' and a pagination control.

Text	Source	City	Language	Sentiment
@javi_Sancho_ La final de la ncaa de baloncesto	Twitter for Android		es	Neutral
@jjmbt32 Enhorabuena, pocos viajes me he pegado yo detrás de la minitere, que estaba en la selección de baloncesto	Twitter for Android		es	Positive
@joseamartinmdv Es ese tipo de jugador que lo ves y no parece jugador profesional de baloncesto, siempre te hace lo mismo y te la acaba metiendo. Y	Twitter for Android		es	Negative
A pesar de no ser jugador profesional, Imbroda tuvo una extensa carrera en los banquillos. Entrenó a la selección y también a la sección de baloncesto del Real	TweetDeck		es	Neutral
Aquí os dejo una propuesta de calentamiento pre-partido en baloncesto: -Cuando no tenemos pista- Activación socioafectiva Activación metabólica	Twitter Web App		es	Neutral

Ilustración 5.18. Interfaz datasets creados

Esta vista (Ilustración 5.18) muestra los tweets contenidos en el dataset seleccionado en el desplegable colocado en la parte superior izquierda. Con los botones “Delete” y “Export” podemos eliminar o exportar el dataset seleccionado. El botón “Create” nos abre el formulario de creación de un dataset (Ilustración 5.16) y el botón “Import” nos permite importar un dataset previamente creado y exportado.

En el listado de tweets nos aparece el texto del tweet, desde dónde se ha publicado, ciudad desde donde se ha publicado (si se ha publicado con la localización activada), el idioma del tweet y por último si expresa un sentimiento positivo, negativo o neutral.

Vista de las estadísticas de un dataset

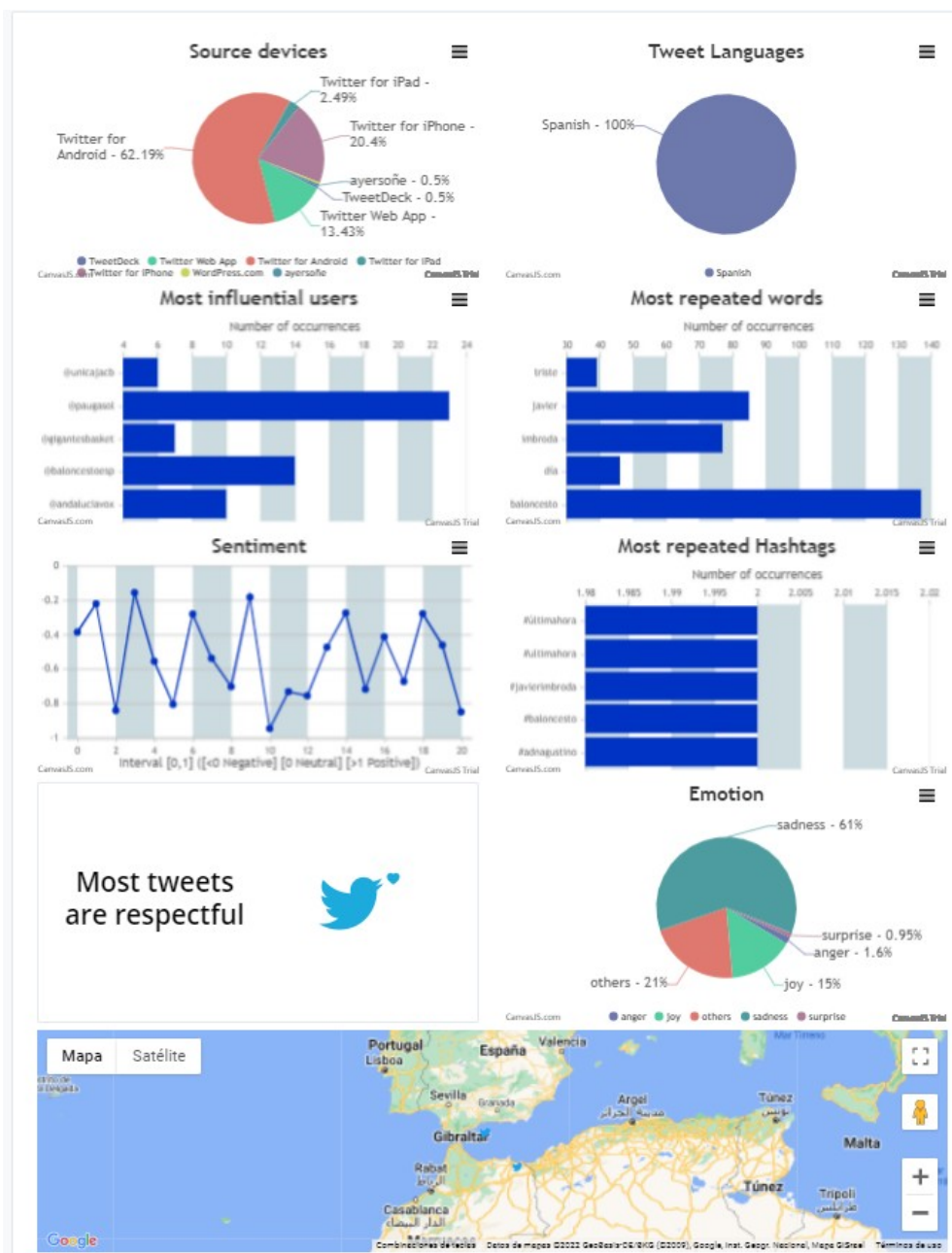


Ilustración 5.18. Estadísticas de un dataset

Esta es la vista (Ilustración 5.18) en la que se muestran las estadísticas recuperadas del dataset, en la parte superior izquierda tenemos un desplegable para poder elegir de qué dataset se quiere ver las estadísticas. Las gráficas son las mismas ya explicadas anteriormente, con la diferencia de que los sentimientos (Sentiment) se representan mediante una gráfica de líneas que nos permite ver la progresión del sentimiento a lo largo del tiempo. En esta vista también se muestra un mapa en caso de que algún tweet tenga localización, y marca esta con el icono de Twitter.

6 IMPLEMENTACIÓN

En esta sección se detallan las fases de la implementación, así como las tecnologías que utiliza el software y las herramientas usadas para su implementación.

6.1 Aplicación de SCRUM

6.1.1 Sprint 1

En el primer sprint se realizó una de las partes más importantes de los proyectos software como es el Análisis de Requisitos, en esta parte se respondieron junto a los tutores a las siguientes preguntas:

- **¿Se debe realizar la aplicación web con algún Framework determinado?**

No, la recomendación es que fuese algún Framework de Python por su facilidad de integración con las herramientas de PLN existentes, se dejó a elección del alumno y finalmente se optó por Flask¹².

Antes de optar por Flask, se hizo un estudio comparativo de los distintos frameworks [20] de Python, los más usados son Django, Flask, Pyramid y Web2py. Estos dos últimos no son tan populares y tienen menor documentación y comunidad, por ello finalmente el estudio consistió en comparar Django y Flask [21].

Obviamente, no podemos decir que se haya elegido Flask por ser mejor que Django, se eligió esta alternativa porque es mucho más flexible:

- Debemos entender que Flask surgió 5 años después que Django, ofreciendo novedades y alternativas al Modelo Vista Controlador.
 - No se tiene experiencia con ninguno de los dos frameworks por lo que se ha preferido Flask que instala extensiones de forma fácil y rápida. Además, según su comunidad, es una mejor opción para personas que desarrollan en Python desde cero.
 - La velocidad media de Flask es superior a la de Django.
 - Ofrece su propio servidor web de desarrollo, por lo que no es necesario la instalación de un servidor web para usar la aplicación (para un entorno de producción, debería de usarse un servidor web).
- **¿Los usuarios de la aplicación web deben registrarse o podrán entrar sin una cuenta previamente creada?**

No, no es necesario que los usuarios tengan que estar registrados.

12 <https://flask.palletsprojects.com/en/2.0.x/>

- **¿Qué tipo de análisis realizará la aplicación sobre los tweets recuperados?**

El propósito del análisis es obtener información sobre temas de actualidad en la red Social de Twitter. En el análisis se debe aplicar técnicas de PLN, obtener las tendencias que discuten los usuarios por localización. Se podrá recuperar tweets en tiempo real.

- **¿Qué estadísticas debe ofrecer?**

- Clasificar tweets geográficamente, por palabras clave, menciones de usuarios o hashtags.
- Análisis de la polaridad, emociones expresadas y detección de lenguaje de odio en una muestra de tweets.
- Se deben aplicar otro tipo de técnicas de PLN estudiadas en el grado.

- **¿Qué debe entregarse al finalizar el proyecto?**

- Documentación del TFG (presente documento).
- Código fuente y ejecutable del código.
- Video demostrativo del funcionamiento de la aplicación.
- Manual de usuario.

6.1.2 Sprint 2

El segundo sprint supuso el comienzo de la aplicación. Para ello se eligió una [plantilla](#) de bootstrap que se aproximase a la idea preconcebida de la interfaz del proyecto.

Otra parte fundamental fue la realización de un tutorial¹³ de Flask para saber empezar el desarrollo del proyecto, una vez concluido dicho tutorial, se añadió la plantilla de bootstrap escogida y se comenzó su adaptación. Para el diseño del logotipo (Ilustración 6.1) y del icono (Ilustración 6.2) de la aplicación se usó Adobe Photoshop y el resultado fue:



Ilustración 6.1. Logotipo de la aplicación

13 <https://j2logo.com/tutorial-flask-espanol/>



Ilustración 6.2. Icono de la aplicación

6.1.3 Sprint 3

Durante el tercer sprint se implementaron las funcionalidades básicas de la aplicación, es decir, se aseguró que los requisitos funcionales se cumpliesen. Una vez acabado el sprint se validaron las funcionalidades disponibles de la aplicación para determinar si se necesitase alguna más. Por parte de los tutores del proyecto se requirió que la aplicación pudiese almacenar y trabajar con varios datasets simultáneamente, ya que, hasta este sprint, se había implementado que el usuario solo podría tener un dataset en el sistema. Este cambio supuso un gran esfuerzo de implementación porque la lógica de la aplicación no estaba preparada para él.

También se requirieron varios cambios para mejorar el aspecto visual de la aplicación y disminuir la carga cognitiva del usuario, los más importantes fueron:

- En la pantalla principal, sustituir el carrusel de imágenes por un listado de las principales funcionalidades de la aplicación, se propuso el cambio porque el texto sobre las imágenes no podría verse muy bien.
- En la vista del dataset, aparecían todos los datos guardado del dataset, se propuso que se mostrase un resumen del mismo. También, se suprimieron los datos del dataset que no han sido necesarios para los análisis implementados.
- La puntuación de la polaridad de los tweets, es decir, la forma de determinar si un tweet es positivo, negativo o neutral se expresaba con un número en un intervalo $[0,5]$, se sugirió que se cambiase por una barra de progreso porque el usuario podría no entender el número.

6.1.4 Sprint 4

Este sprint se centró en resolver los problemas detectados por parte de los tutores en la revisión del anterior sprint, además, se incorporó la API de Google Maps para que las localizaciones se determinasen mediante marcadores en mapas proporcionados por la propia API. Este sprint, supone el final de la fase de implementación de la aplicación siempre y cuando no se detecten nuevas mejoras ni fallos.

6.1.5 Sprint 5

Una vez terminada la fase de implementación, queda completar la documentación restante, comprobar que todo el sistema funcione, realizar el ejecutable, manual de usuario y video demostrativo solicitado por parte de los tutores.

6.2 Tecnologías

6.2.1 Python

Como lenguaje de programación usado se ha usado Python 3.9.5, se trata de un lenguaje de alto nivel y orientado a objetos. Se ha elegido este lenguaje de programación porque la mayoría de herramientas de IA, y concretamente, de PLN se han desarrollado para este lenguaje, además, ya se contaba con experiencia adquirida en la asignatura de Procesamiento del Lenguaje Natural cursada durante el grado en la cual se usan corpus y herramientas como nltk, spacy, textblob, etc. Que se han aplicado en mayor o menor medida en el desarrollo del proyecto.

6.2.2 Flask

Flask es un framework desarrollado en Python para desarrollo web. Aunque se podría realizar el backend en python para aplicar técnicas de PLN y el frontend en cualquier otro framework, se ha considerado que mejor se unifiquen las dos partes en el mismo lenguaje de programación para simplificar la implementación. La versión usada es Flask 2.0.2.

6.2.3 API Google Maps

Para interactuar con mapas seguramente la opción más usada sea usar la API de Google Maps, esta herramienta proporciona bastantes posibilidades, pero en el proyecto se ha usado para ubicar los tweets recuperados en un mapa usando las coordenadas proporcionadas por la API de Twitter. También se ha usado para ubicar un marcador que se usa como punto de referencia para buscar tweets publicados cerca de esa ubicación. Para usarla debemos registrarnos en Google Cloud Console, introducir los datos bancarios, activar esta API e incorporar las claves en el código JavaScript que comunicará el usuario con la API.

6.2.4 Tweepy

Se trata de una librería de Python que permite acceder a la API de Twitter. Para usar esta API, debemos de tener una cuenta de Twitter y registrarnos en el portal de desarrolladores de Twitter, durante el registro nos harán una serie de preguntas para que Twitter sepa el propósito de la aplicación a implementar y dependiendo de nuestras necesidades deberemos de esperar a que Twitter de el visto bueno a la aplicación. De esta forma, por ejemplo, un investigador de la universidad puede acreditarse y obtener más recursos de la API de forma gratuita. Una vez activada la cuenta, obtendremos las credenciales de acceso y uso de la API que hay que incorporar para usarla. La versión de Tweepy utilizada es la 4.3.0.

6.2.5 PySentimiento

Es un conjunto de herramientas de PLN desarrolladas para Python por investigadores de la Universidad de Buenos Aires. Actualmente, con esta herramienta podemos realizar análisis de sentimiento, análisis de emociones y detección de discurso de odio, todo esto en español e inglés.

Esta herramienta está continuamente mejorando sus modelos de aprendizaje automático e incorpora su propio preprocesador de textos, lo que hace de ella una herramienta simple y a la vez potente.

6.2.6 Bootstrap

Es un framework frontend creado por Twitter en 2010. Se compone de un conjunto de clases CSS que bien usadas garantizan un sitio web responsivo y visualmente atractivo. Bootstrap incorpora una documentación¹⁴ útil y concisa con multitud de ejemplos del uso de sus clases. En internet podemos encontrar infinidad de temas y plantillas de Bootstrap para usarlas como punto de partida para nuestra aplicación. La versión de bootstrap utilizada es la 4.

6.2.7 JQuery

Para el uso de ciertos plugins de bootstrap ha sido necesario incorporar JQuery 1.9.1. JQuery es una librería que usa javascript, su principal motivación es que los desarrolladores no tengan que preocuparse de en qué navegador se ejecutará el código. Además proporciona una codificación más condensada y legible que si se usase javascript únicamente.

¹⁴ <https://getbootstrap.com/docs/4.1/getting-started/introduction/>

6.2.8 DataTables

Es un plugin de JQuery que añade funcionalidades a las tablas implementadas en HTML. Se ha usado en los listados de tweets, de tal forma que simplemente instanciando una tabla como DataTable podemos realizar búsquedas de cadenas en la tabla y paginación de la tabla.

6.2.9 SweetAlerts

Es un plugin de javascript que proporciona alertas responsivas. La alternativa tradicional es incorporar un modal en el HTML, evidentemente, diseñar un popup mediante HTML es mucho más flexible, pero para ventanas de diálogo, confirmación, etc. SweetsAlerts permite generar Modals de forma mucho más rápida. La versión utilizada es la 2.11.

6.3 Herramientas

6.3.1 PyCharm IDE

Como interfaz de desarrollo se ha elegido PyCharm 2021.1.2. Es una interfaz de desarrollo simple pero muy útil para desarrollar en Python, además también ha sido útil para el desarrollo web ya que, como la mayoría de IDEs, pueden usarse para HTML, javascript, JSON, etc.

6.3.2 Photoshop CS6

Es uno de los programas de diseño gráfico más utilizados, en este proyecto se ha utilizado para el diseño del logotipo, icono y retoque de alguna imagen incluida en la aplicación.

6.3.3 Gantt Project

Es un software para la gestión de proyectos. Su función principal es la realización de diagramas de Gantt para obtener una planificación temporal de las diferentes tareas en las que se ha descompuesto el proyecto. Una vez hecho el diagrama de Gantt podemos generar automáticamente el diagrama PERT. También permite añadir recursos a las tareas lo que nos ayuda a ver cuántos recursos son necesarios y calcular los costes.

7 PRUEBAS Y EVALUACIÓN DE USABILIDAD

Antes de entregar cualquier producto software, debe asegurarse que el producto que se va a entregar satisface las necesidades del cliente y se ajusta a lo que ha pedido. Para ello. Para poder testear la aplicación correctamente se han hecho dos tipos de pruebas, la primera trata de asegurarse que las funcionalidades requeridas están implementadas y funcionando y la segunda trata de recopilar el punto de vista sobre la aplicación de usuarios escogidos de manera aleatoria.

7.1 Pruebas de validación

En la tabla siguiente se exponen un listado pruebas que deberían de garantizar el cumplimiento de requisitos junto a su comprobación de funcionamiento:

Prueba	Comprobación
Los botones de la barra de navegación funcionan y el desplegable con las regiones disponibles funciona.	Sí.
Se pueden obtener las estadísticas de una tendencia a partir de una región escogida al azar. El número de tweets recolectados se corresponde con el indicado.	Sí.
Se pueden obtener las estadísticas de una tendencia a partir de una palabra clave. El número de tweets recolectados se corresponde con el indicado.	Sí.
Crear nuevo dataset por geolocalización.	Sí.
Crear nuevo dataset por palabra clave.	Sí.
Parar la recopilación de tweets de un dataset.	Sí.
Eliminar dataset.	Sí.
Importar dataset.	Sí.
Exportar dataset.	Sí.
Visualización de estadísticas (al menos un tweet con localización para que se muestre el mapa).	Sí.
Visualización de estadísticas (tweets sin localización para que oculte el mapa).	Sí.
Permitir cambio de dataset a la hora de visualizar estadísticas.	Sí.

7.1. Tabla: Pruebas de validación

7.2 Pruebas de usabilidad y experiencia de usuario

Las pruebas de usabilidad y UX se ha realizado mediante un cuestionario tras la prueba de la aplicación por parte de un grupo de personas, han probado la aplicación un total de cinco personas las cuales han respondido a las siguientes cuestiones:

Dificultad de instalación de la aplicación

5 respuestas

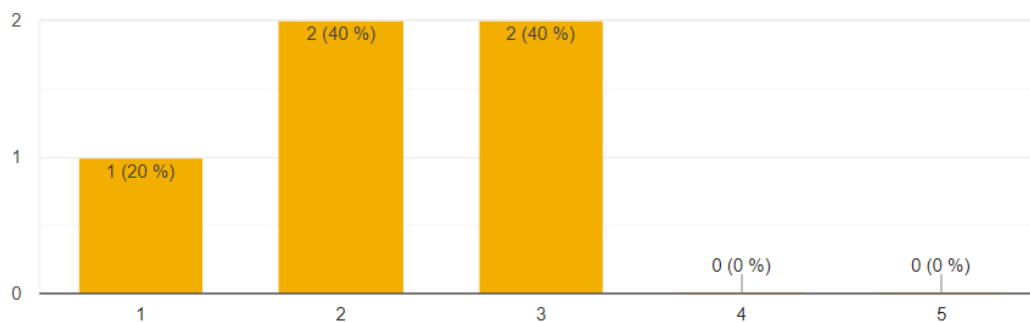


Ilustración 7.1. Encuesta – Dificultad de instalación de la aplicación

Este punto del cuestionario (Ilustración 7.1) pretende ver si el ejecutable e instrucciones de instalación facilitadas son lo suficientemente sencillas de utilizar. La puntuación es 1 → Muy fácil y 5 → Muy difícil, por lo tanto se puede ver que la dificultad de instalación es Media-Sencilla.

Dificultad de uso de la aplicación

5 respuestas

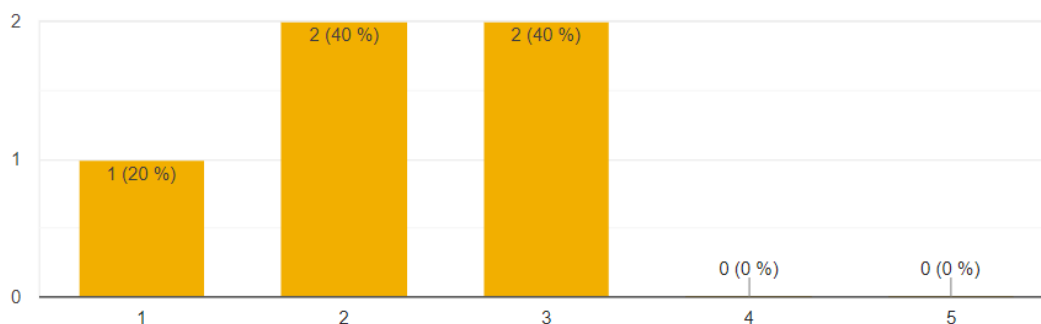


Ilustración 7.2. Encuesta – Dificultad de uso de la aplicación

La dificultad de uso de la aplicación es muy importante pues no se requieren conocimientos previos para su uso, es decir, cualquier persona podría usarla. Los usuarios encuestados han determinado (Ilustración 7.2) que la dificultad de uso es Media-Sencilla.

Tiempo de respuesta de la aplicación

5 respuestas

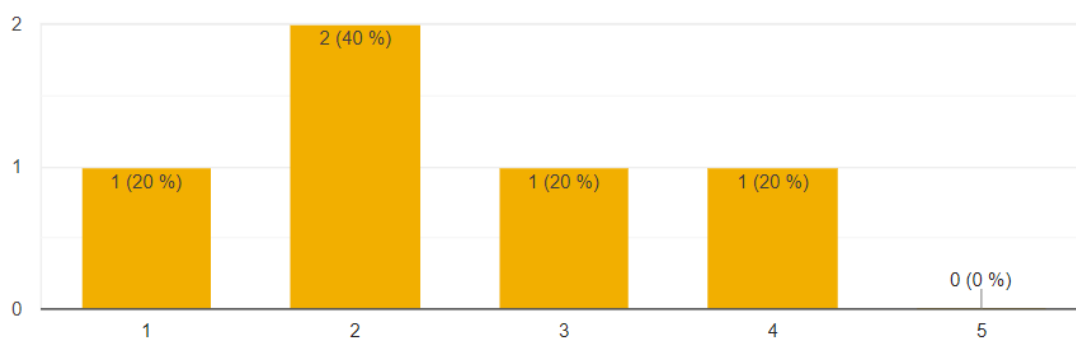


Ilustración 7.3. Encuesta – Tiempo de respuesta de la aplicación

Puntuación: 1 → Muy rápido, 5 → Muy lento. Los resultados (Ilustración 7.3) aquí han estado demasiado repartidos para sacar una conclusión clara, al menos podemos decir que la mayor parte de los usuarios encuestados no piensan que la aplicación sea lenta.

Número de errores producidos durante su uso

5 respuestas

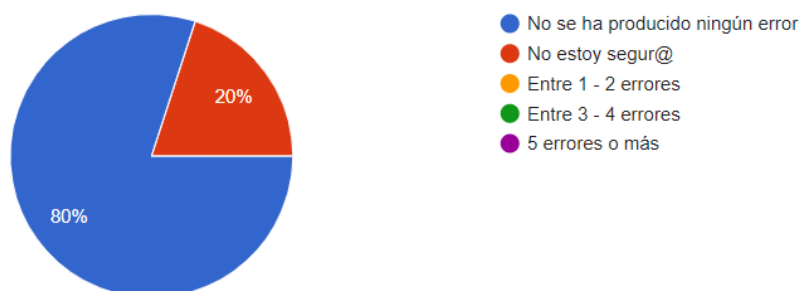


Ilustración 7.4. Encuesta – Número de errores producidos durante su uso

El 80% de usuarios (Ilustración 7.4) no han percibido ningún error en la aplicación, y el 20% restante no está seguro.

¿Recomendarías la aplicación?

5 respuestas

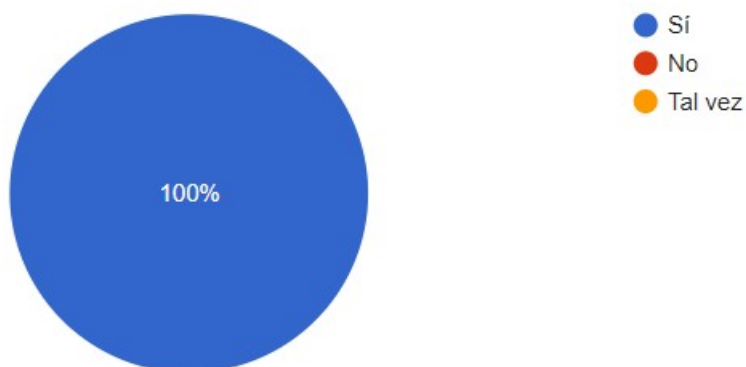


Ilustración 7.5. Encuesta – Recomendación de la aplicación

Todos los usuarios encuestados recomendarían la aplicación (Ilustración 7.5).

¿Qué nota le pondrías a la aplicación?

5 respuestas

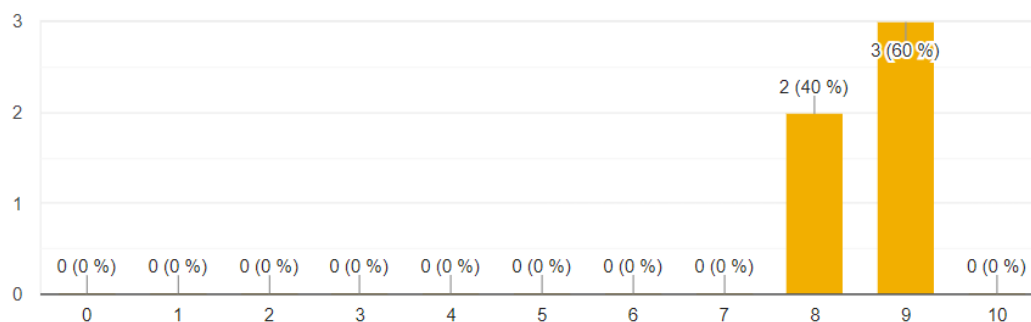


Ilustración 7.6. Encuesta – Puntuación de la aplicación

Puntuación: 0 → Muy mala, 10 → Muy buena. Todos los usuarios encuestados han puesto una nota bastante buena a la aplicación (Ilustración 7.6).

8 CONCLUSIONES

En esta sección se tratan las conclusiones obtenidas tras la realización del proyecto, así como las mejoras que se podrían incluir si se decidiese aumentar el alcance del mismo.

8.1 Valoración personal

En mi opinión, TrendsAnalytics es una herramienta bastante novedosa y sencilla de usar en comparación a sus rivales en el mercado. La inclusión de técnicas de PLN en estos tipos de software debería ser prácticamente una obligación, ya que como se ha dicho, nos ofrece unas posibilidades imposibles de conseguir de otra forma.

Otro punto fuerte de mi aplicación diría que la capacidad de personalización de las búsquedas, no está presente en otras aplicaciones, es decir, otras aplicaciones solo permiten obtener información de las tendencias actuales, o de alguna regiones, pero TrendsAnalytics permite realizar búsquedas de cualquier palabra o localización.

Aprovechando que la aplicación se ejecuta en un entorno local, no tenemos que tener tanto en cuenta la reducción del cómputo que realiza el servidor, ya que si se usase un hosting de pago, la captura y procesamiento de una gran cantidad de tweets no sería barata.

Respecto a la detección de lenguaje de odio, es un aspecto, no solo de la aplicación, sino del PLN que deja mucho que desear, y en mi opinión, pasará mucho tiempo hasta que se consiga una herramienta realmente precisa. El lenguaje del odio es un tema muy subjetivo, porque un tweet es un texto demasiado corto y no tenemos ayuda de entonación para deducir la intención con la que se dicen las palabras, a cualquier persona le costaría identificarlo, por lo tanto a la IA también le cuesta.

Personalmente, tras la realización del proyecto, he adquirido conocimientos sobre el Framework de Flask, a parte de mejorar mis habilidades de Javascript, Python, JQuery, etc. Respecto a las técnicas de PLN, tenía ya una cierta base adquirida en la asignatura del grado cursado, no obstante, es fascinante descubrir qué herramientas hay disponibles y lo “sencillo” que es utilizarlas. Aun así, creo que todo lo relacionado con el PLN es aún una tecnología que no se está explotando todo lo que se pudiera, y creo que se debería de hacer más para dar a conocer su potencial a los directivos de las tecnológicas para que poco a poco apuesten seriamente por el desarrollo e inclusión de nuevas técnicas en sus productos.

8.2 Posibilidades de mejora

La principal mejora, por no decir obligación para el uso de esta aplicación por el público en general, sería el registro de usuarios para asociarles una cuota como límite para recuperar tweets, ya que no se puede ofrecer un uso ilimitado de una aplicación que demanda tanto cómputo de forma gratuita.

Como ya se ha mencionado, la aplicación puede trabajar en dos idiomas (Español e Inglés), no obstante, el código fuente está preparado para incluir fácilmente más idiomas siempre y cuando se dispongan de herramientas similares a PySentimiento para dichos idiomas. Algunas de las herramientas de NLTK utilizadas también son dependientes del idioma, pero este problema sería menor, ya que proporcionando el identificador del lenguaje que se capta junto al tweet, p.ej: “es”, “en”, “ja”... Podemos utilizar dicha herramienta en multitud de idiomas.

Otra opción interesante que mientras se recopilan tweets para un dataset, notificar al usuario, ya sea por correo electrónico, sms, o similar, de un cambio de tendencia en las estadísticas generadas, por ejemplo, si estoy recuperando tweets continuamente sobre mi marca, me interesaría conocer si en algún momento, los usuarios empiezan a hablar mal de mi marca. Además podríamos emplear otras técnicas de PLN [24] como:

- Búsqueda de respuestas utilizando la información recogida en un dataset.
- Generación de resúmenes de la temática de una tendencia.
- Etiquetado de imágenes para poder describir textualmente el contenido de una imagen adjuntada a un tweet.
- Extracción de entidades para saber de qué o quién se está hablando.

ANEXO I – MANUAL DE INSTALACIÓN

A continuación se exponen los requisitos y pasos a seguir para la instalación y la ejecución de la aplicación, el primer paso es descargar el proyecto desde el [Gitlab](#) de la Universidad de Jaén, la carpeta descargada contiene la siguiente información (Ilustración AI.1):

Nombre	Fecha de modificación	Tipo	Tamaño
.idea	11/04/2022 18:06	Carpeta de archivos	
app	11/04/2022 18:06	Carpeta de archivos	
.gitignore	11/04/2022 18:06	Documento de tex...	3 KB
executable	11/04/2022 18:06	Archivo por lotes ...	1 KB
README	11/04/2022 18:06	Documento de tex...	1 KB
requirements	11/04/2022 18:06	Documento de tex...	2 KB
run	11/04/2022 18:06	Python File	1 KB

Ilustración AI.1: Contenido carpeta proyecto

La carpeta contiene un script “run.py” que arranca la aplicación, pero este solo funcionará si ya tenemos los requisitos instalados. Por ello existe el “**executable.bat**” que comprueba si se tienen los requisitos especificados en “requirements.txt”, en caso de que no estén instalados, los descarga y los instala, si ya están instalados, llama a “run.py”.

Los únicos **prerrequisitos** para la ejecución de la aplicación es tener instalado Python y Pip, concretamente para la elaboración de este proyecto se han usado las versiones **Python 3.9.5** y **pip 22.0.4**.

ANEXO II – MANUAL DE USUARIO

Una vez ejecutada la aplicación, podemos acceder a ella en el siguiente enlace: <http://127.0.0.1:5000/>, se nos abrirá la pantalla principal, en ella tendremos acceso a las distintas funcionalidades mediante la barra de navegación (Ilustración AII.1):



Ilustración AII.1. Barra de navegación

1. Acceso a la pantalla principal.
2. Acceso a la funcionalidad para buscar tweets por palabra clave.
3. Acceso a la funcionalidad para la vista de visualización, creación y borrado de conjuntos de datos captados en tiempo real.
4. Acceso a la vista de visualización de las estadísticas generadas a partir de los conjuntos de datos captados en tiempo real.
5. Listado y acceso a búsqueda de tendencias por región.

Las funcionalidades de búsqueda por palabras clave y búsqueda de tendencias por regiones son similares (Ilustración AII.2 e Ilustración AII.3):

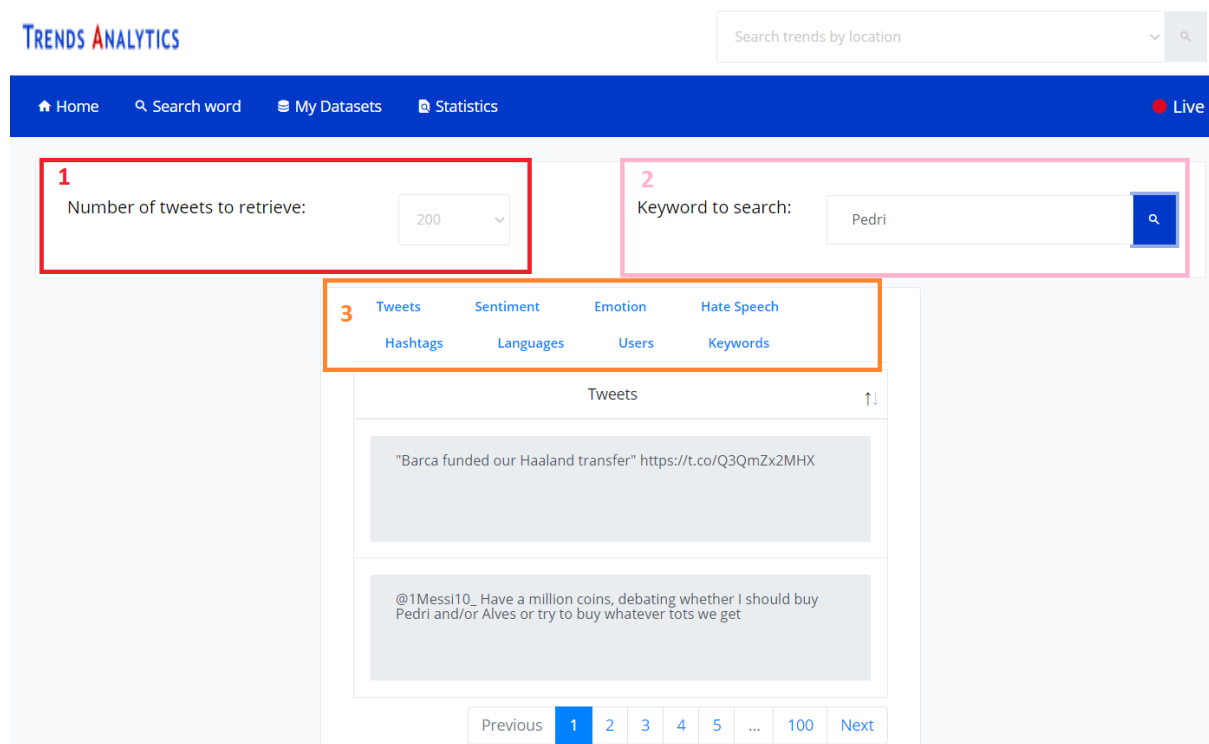


Ilustración AII.2. Vista búsqueda por palabras clave

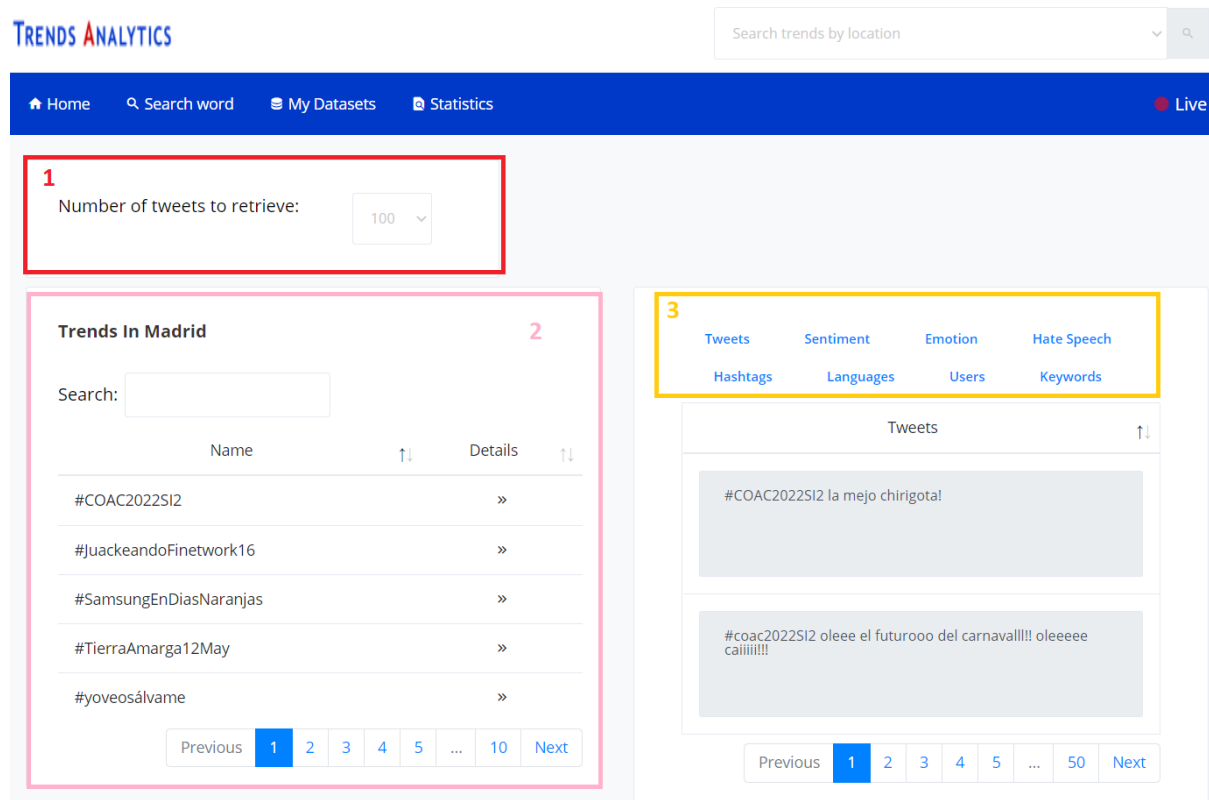


Ilustración AII.3. Vista búsqueda de tendencias por región

1. Campo para indicar el número de tweets a recuperar, si el usuario introduce un número alto, el análisis de los tweets captados puede tardar unos minutos.

2. Campo para introducir las palabras clave, en caso de búsqueda de tendencias por regiones, este campo se cambia por el listado de tendencias actuales en esa región.
3. Botones para visualizar las distintas estadísticas generadas.

En la funcionalidad de “My Datasets” se muestra un resumen de los conjuntos de datos captados en tiempo real generados (Ilustración AII.4):

The screenshot displays the 'My Datasets' section of the Trends Analytics application. At the top, there is a search bar and a navigation bar. The main area is divided into three numbered sections:

- 1**: A dropdown menu for selecting a dataset, currently showing 'Baloncesto.csv'.
- 2**: A set of buttons for 'Create', 'Delete', 'Export', and 'Import'.
- 3**: A table of search results for the selected dataset. The table has columns for Text, Source, City, Language, and Sentiment. It shows five entries, all with negative sentiment (red sad face icons).

Below the table, it says 'Showing 1 to 5 of 201 entries' and a pagination bar with 'Previous', '1', '2', '3', '4', '5', '...', '41', and 'Next'.

Ilustración AII.4. Vista My Datasets

1. Desplegable para seleccionar el dataset con el que trabajar.
2. Botones de creación, borrado, exportación e importación de datasets.
3. Vista de resumen del dataset seleccionado.

En la vista de estadísticas de los datasets, se muestra distintas gráficas generadas a partir del análisis realizado sobre los tweets (Ilustración AII.5):

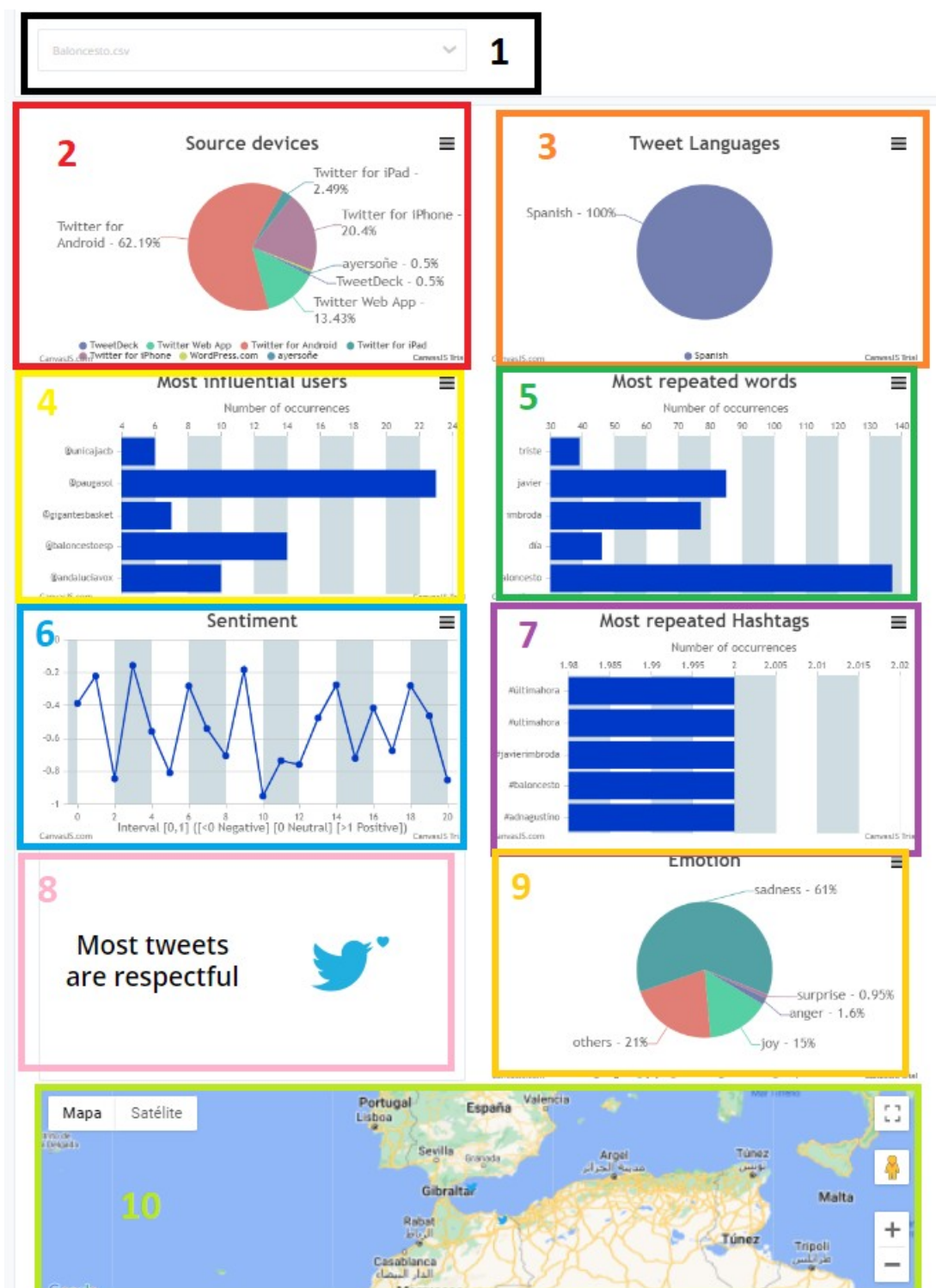


Ilustración AII.5. Vista estadísticas de los datasets

1. Desplegable para seleccionar de qué datasets se quiere visualizar las estadísticas.
2. Dispositivos o plataformas desde los que se publicó los tweets captados.

3. Idiomas de los tweets captados.
4. Usuarios más influyentes.
5. Palabras más repetidas.
6. Sentimiento de los usuarios a lo largo del tiempo. ($<1 \rightarrow$ Negativo, $= 0 \rightarrow$ Neutro, $>1 \rightarrow$ Positivo).
7. Hashtags más repetidos.
8. Lenguaje de odio y misoginia captado.
9. Emociones captadas.
10. Geolocalización de los tweets.

ANEXO III – ÍNDICE DE ILUSTRACIONES

Portada.....	1
Firma.....	1
Usuarios RRSS por edad.....	6
Ciclo de vida SCRUM.....	10
Tablero GitLab Issues.....	11
Genealogía redes sociales.....	14
Dashboard herramienta de análisis HootSuite.....	19
GetDayTrends evolución de tendencia.....	20
Tweeplers Hashtags List.....	20
Tweeplers Real Time Tweets Map.....	21
Diagrama de Gantt I.....	28
Diagrama de Gantt II.....	30
Diagrama PERT.....	31
Planes Tweepy.....	32
Diagrama de caso de uso (General).....	34
Diagrama de caso de uso (Ver estadísticas de las tendencias actuales por región).....	34
Diagrama de caso de uso (Obtener estadísticas actuales a partir de una palabra clave).....	35
Diagrama de caso de uso (Crear Dataset).....	36
Diagrama de caso de uso (Visualización, borrado y descarga de datasets).....	37
Diagrama de secuencia (Ver estadísticas de las tendencias actuales por región).....	38
Diagrama de secuencia (Obtener estadísticas actuales a partir de una palabra clave).....	39
Diagrama de secuencia (Creación de un Dataset y visualización de sus estadísticas).....	40
Diagrama diseño Frontend.....	42
Diagrama diseño Backend.....	43
Interfaz página principal.....	44
Interfaz listado de regiones.....	45
Listado de tendencias de una región.....	45

Interfaz búsqueda de tweets por palabra clave.....	46
Interfaz listado de tweets.....	46
Interfaz sentimiento.....	47
Interfaz HateSpeech.....	47
Interfaz emociones.....	48
Interfaz hashtags.....	48
Interfaz idiomas.....	49
Interfaz usuarios.....	49
Interfaz palabras con más ocurrencias.....	50
Interfaz mis datasets creados (vacío).....	50
Interfaz formulario de creación de un dataset por palabra clave.....	51
Interfaz formulario de creación de un dataset por geolocalización.....	51
Interfaz datasets creados.....	52
Estadísticas de un dataset.....	53
Logotipo de la aplicación.....	55
Icono de la aplicación.....	56
Encuesta – Dificultad de instalación de la aplicación.....	62
Dificultad de uso de la aplicación.....	63
Encuesta – Tiempo de respuesta de la aplicación.....	63
Encuesta – Número de errores producidos durante su uso.....	64
Encuesta – Recomendación de la aplicación.....	64
Encuesta – Puntuación de la aplicación.....	65
Contenido carpeta proyecto.....	68
Barra de navegación.....	69
Vista búsqueda por palabras clave.....	70
Vista búsqueda de tendencias por región.....	70
Vista My Datasets.....	72
Vista estadísticas de los datasets.....	73

ANEXO IV – ÍNDICE DE TABLAS

Tabla: Comparativa de analizadores de tendencias.....	22
Tabla: Pruebas de validación.....	62

Bibliografía

- [1] Fernández, R. F. (2021). Edad de los usuarios de redes sociales. Statista. <https://es.statista.com/estadisticas/1260093/redes-sociales-porcentaje-de-usuarios-por-edad-en-espana/>
- [2] Kemp, S. (2021). Digital 2021: Global Overview Report. DataReportal – Global Digital Insights. Recuperado 27 de febrero de 2022, de <https://datareportal.com/reports/digital-2021-global-overview-report>
- [3] Santos, Diego (2022). 109 datos y estadísticas sobre redes sociales para 2022. <https://blog.hubspot.es/marketing/estadisticas-redes-sociales>
- [4] Schwaber, Ken. Sutherland, Jeff (2020). Guía de SCRUM. <https://scrumguides.org/scrum-guide.html>
- [5] M. Pérez-Wiesner, M.P. Fernandez-Martín, F. López-Muñoz (2014). El fenómeno de las redes sociales: evolución y perfil del usuario
- [6] Anna Ujwary-Gil (2021). Social Network Analysis
- [7] Stanley WasserMan, Katherine Faust (2013). Social network analysis.
- [8] Juan Antonio Lloret Egea (2020). The states of Artificial Intelligence.
- [9] Rodríguez, C., & Dorado, R. (2015). Revista Ontare, 3(1). ¿Por qué implementar Scrum?.
- [10] Martín, S. M. (2021). Redes sociales más utilizadas. mkparadise. Recuperado 28 de febrero de 2022, de <https://mkparadise.com/redes-sociales-mas-utilizadas>
- [11] Boyd, D. M., & Ellison, N. B. (2007). Social Network Sites: Definition, History, and Scholarship. Journal of Computer-Mediated Communication, 13(1), 210–230. <https://doi.org/10.1111/j.1083-6101.2007.00393.x>
- [12] Crua, C. (2020, 3 noviembre). ¿Cómo son los usuarios de Twitter y cómo se comportan con las marcas? Thinking for Innovation. <https://www.iebschool.com/blog/estadisticas-usuarios-twitter-como-son-redes-sociales/>
- [13] Colaboradores de Wikipedia. (2022, 24 febrero). Twitter. Wikipedia, la enciclopedia libre. <https://es.wikipedia.org/wiki/Twitter>
- [14] Ahlgren, M. (2022, 5 enero). Más de 50 estadísticas y datos de Twitter para 2022. Website Rating. <https://www.websiterating.com/es/research/twitter-statistics/>
- [15] Colaboradores de Wikipedia. (2022a, febrero 5). Análisis de redes sociales. Wikipedia, la enciclopedia https://es.wikipedia.org/wiki/An%C3%A1lisis_de_redes_sociales#Historia
- [16] Education, I. C. (2021, 17 agosto). Natural Language Processing (NLP). IBM. <https://www.ibm.com/cloud/learn/natural-language-processing>
- [17] Pauli, P. A. P. (2019, julio). Análisis de sentimiento. Comparación de algoritmos predictivos y métodos utilizando un lexicón español. <https://ri.itba.edu.ar/bitstream/handle/123456789/1782/Proyecto%20Final.pdf?>

[sequence=1&isAllowed=y#:~:text=El%20An%C3%A1lisis%20de%20Sentimientos%20\(AS,acerca%20de%20un%20determinado%20tema.](#)

- [18] Berzal, F. B. (s. f.). Especificación de requerimientos. UGR. <https://elvex.ugr.es/idbis/db/docs/design/2-requirements.pdf>
- [19] Colaboradores de Wikipedia. (2020). WOEID. Wikipedia, la enciclopedia libre. <https://es.wikipedia.org/wiki/WOEID>
- [20] Delgado, D. O. (2020). Los 4 mejores frameworks para aplicaciones de Python. OpenWebinars.net. <https://openwebinars.net/blog/los-4-mejores-frameworks-para-aplicaciones-de-python/>
- [21] Rodríguez, X. (2020). Django vs Flask. OpenWebinars.net. <https://openwebinars.net/blog/django-vs-flask/>
- [22] Felipe, P. (2020) Detección de discurso de odio en redes sociales. https://addi.ehu.es/bitstream/handle/10810/48827/Detecci_n_de_discurso_de_odio_en_RRSS.pdf?sequence=1
- [23] Plaza-Del-Arco, F. M., Molina-Gonzalez, M. D., Urena-Lopez, L. A., & Martin-Valdivia, M. T. (2021). A Multi-Task Learning Approach to Hate Speech Detection Leveraging Sentiment Analysis. IEEE Access, 9, 112478–112489. <https://doi.org/10.1109/access.2021.3103697>
- [24] Velez, J. I. (2016). Integración de Técnicas de Procesamiento de Lenguaje Natural a través de Servicios Web. Universidad Nacional del Centro de la Provincia de Buenos Aires. <http://alejandrorage.com.ar/files/advising/2016-thesis-velez&ramos.pdf?i=1>