

**MÁSTER UNIVERSITARIO EN SEGURIDAD DE LAS
TECNOLOGÍAS DE LA INFORMACIÓN Y LAS
COMUNICACIONES**

TRABAJO FIN DE MÁSTER

**Análisis OSINT, un enfoque práctico.
Huella Digital.**

**Autor
Jorge Generelo Gimeno**

**Director del Trabajo Fin de Máster
Carlos Pintos Teigeiro**

CURSO 2020-2021



RESUMEN

Hoy en día el número de usuarios y dispositivos conectados a internet se ha visto aumentado de manera considerable con el auge de las nuevas tecnologías y el uso masivo de las redes sociales. Cada una de las actividades realizadas sobre la red deja un rastro o registro que recibe el nombre de “huella digital” y que puede aportarnos información relevante sobre el usuario del dispositivo utilizado para la conexión.

El objeto del presente trabajo, es el análisis de la información que podemos obtener sobre una persona recopilando su huella digital en las redes sociales gracias al uso de metodologías y técnicas OSINT (Open Source INTelligence) haciendo referencia este término a la recolección, análisis y toma de decisiones sobre datos de fuentes disponibles de forma pública.

La primera parte del trabajo comprenderá un análisis sobre el campo OSINT atendiendo tanto a sus orígenes como a su estado actual y sus principales usos. Una vez comprendido el contexto en el que se desarrolla el proyecto, se comentarán las características principales del *framework* implementado cuyo objetivo consiste en unificar el trabajo con herramientas OSINT en un entorno único y que podemos separar en tres partes: una herramienta web de búsqueda de huella digital de un individuo, herramientas OSINT independientes y la distribución de sistema operativo que aglutina todos los elementos comentados.

De igual manera, el proyecto ha permitido establecer una serie de procedimientos a seguir a la hora de recopilar la información de cada una de las redes sociales atendiendo principalmente a los distintos tipos de datos que se pueden obtener de cada una de ellas y como recopilarlos.

Con el *framework* resultado de la realización de este trabajo se ofrece un entorno de trabajo personal o compartido que incluye todos los elementos necesarios para facilitar la investigación en fuentes abiertas permitiendo al usuario final aprovechar las ventajas de las técnicas y metodologías OSINT de una manera cómoda y sencilla.

Como producto final del proyecto, la herramienta y la distribución desarrolladas se publicarán de forma abierta en GitHub para habilitar su uso a todo aquel usuario que lo requiera y continuar con la implementación de nuevas funcionalidades en función de las necesidades que puedan surgir con el uso del *framework*.



ABSTRACT

Nowadays the number of users and devices connected to the Internet has increased considerably with the rise of new technologies and the massive use of social networks. Each of the activities carried out on the network leaves a trace or record that is called "digital footprint" and that can provide us with relevant information about the user of the device used for the connection.

The object of this work is the analysis of the information that we can obtain about a person by collecting their digital footprint on social networks thanks to the use of OSINT (Open Source INTelligence) methodologies and techniques by referring this term to the collection, analysis and decision-making on data from publicly available sources.

The first part of this work will include an analysis of the OSINT field, considering both its origins and its current state and main uses. Once we understand the context in which the project is being developed, we will discuss the main features of the implemented framework whose objective is to unify the work with OSINT tools in a unique environment and which we can separate into three parts: a web tool for searching an individual's fingerprint, independent OSINT tools and operating system distribution that brings together all the commented elements.

Similarly, the project has made it possible to establish a series of procedures to be followed when collecting information from each of the social networks, mainly taking into account the different types of data that can be obtained from each of them and how to collect them.

The resulting framework of this work offers a personal or shared working environment that includes all the necessary elements to facilitate research in open sources allowing the end user to take advantage of the advantages of techniques and OSINT methodologies in a convenient and simple way.

As a final product of the project, the tool and the distribution developed will be published openly in GitHub to enable its use to any user who requires it and continue with the implementation of new functionalities according to the needs that may arise with the use of the framework.



AGRADECIMIENTOS

Agradecer a los docentes de la Universidad Europea de Madrid por las enseñanzas y experiencias compartidas a lo largo de la realización del máster.

Agradezco también a Carlos Pintos por la ayuda, consejos y sobre todo paciencia mostradas en el desarrollo del proyecto final del máster.

Por último, agradezco a familia y amigos por el continuo apoyo recibido tanto a nivel profesional como personal.

ÍNDICE DE CAPÍTULOS Y ANEXOS

1	INTRODUCCIÓN.....	15
2	ESTADO DE LA CUESTIÓN	17
2.1	ORIGEN OSINT	17
2.2	ESTADO OSINT EN LA ACTUALIDAD	18
2.2.1	TIPOS DE INFORMACIÓN DE FUENTES ABIERTAS	19
2.2.2	HERRAMIENTAS Y FRAMEWORKS OSINT	20
2.2.3	PUNTOS FUERTES DEL PROYECTO.....	25
3	DESCRIPCIÓN DEL PROBLEMA	26
4	SOLUCIÓN PROPUESTA	29
4.1	OBJETIVOS	29
4.2	LOGROS A ALCANZAR	30
4.3	METODOLOGÍA	30
4.4	PLANIFICACIÓN	31
5	PRUEBAS Y VALIDACIÓN	34
5.1	ANÁLISIS ENTORNOS DE DESARROLLO APLICACIONES WEB	34
5.1.1	BACKEND: DJANGO REST API.....	35
5.1.2	FRONTEND: REACT COREUI.....	36
5.2	DESARROLLO APLICACIÓN WEB	36
5.2.1	PREPARACIÓN DEL ENTORNO DE TRABAJO	36
5.2.2	TWITTER.....	38
5.2.3	INSTAGRAM.....	41
5.2.4	FACEBOOK.....	45
5.2.5	LINKEDIN	48
5.2.6	BÚSQUEDA DE PERSONAS Y METODOLOGÍA	51
5.3	PREPARACIÓN DE LA DISTRIBUCIÓN	53
5.3.1	CREACIÓN Y CONFIGURACIÓN DE LA MÁQUINA VIRTUAL	54
6	RESULTADOS	56
7	CONCLUSIONES.....	59

8	TRABAJOS FUTUROS	61
9	APÉNDICES	63
9.1	BIBLIOGRAFÍA.....	63
9.2	MANUAL DE INSTALACIÓN	66
9.3	MANUAL DE USUARIO	75
9.4	GUÍA DE SOLUCIONADO DE ERRORES	83
9.5	TECNOLOGÍAS UTILIZADAS	84

ÍNDICE TABLAS

Tabla 1. Modelo base de datos de perfil de Twitter	40
Tabla 2. Modelo base de datos de publicación de Twitter.	41
Tabla 3. Modelo base de datos de perfil de Instagram.....	44
Tabla 4. Modelo base de datos de publicación de Instagram.	44
Tabla 5. Modelo base de datos de perfil de Facebook	48
Tabla 6. Modelo base de datos de perfil de LinkedIn.....	50

ÍNDICE FIGURAS

Figura 1. Interfaz de Usuario tras realizar una búsqueda en Maltego. (Elaboración propia).....	21
Figura 2. Opciones de uso de theHarvester. (Elaboración propia)	21
Figura 3. Resultados de búsqueda en Shodan. (Elaboración propia)	22
Figura 4. Sitio web de OSINT Framework. (jnordine, 2021)	22
Figura 5. Ejemplo de ejecución de Sherlock con el parámetro username. (sherlock-project, 2021)	23
Figura 6. Lanzamiento de herramienta Osintgram. (Datalux, 2021)	23
Figura 7. Panel de control de DumpItBlue+. (Le-tools, 2021)	24
Figura 8. Diagrama de Gantt del proyecto	32
Figura 9. Tecnologías aplicación web. (Garner, 2021)	34
Figura 10. Ventana de Login de la aplicación. (Elaboración propia)	37
Figura 11. API interactiva de Swagger. (Elaboración propia)	38
Figura 12. Ejemplo perfil de Twitter. (Captura del sitio web de Twitter)	38
Figura 13. Ejemplo de tweet. (Captura del sitio web de Twitter)	39
Figura 14. Visualización de tweets en la aplicación web. (Elaboración propia) ..	41
Figura 15. Ejemplo perfil de Instagram. (Captura del sitio web de Instagram) ..	42
Figura 16. Ejemplo publicación de Instagram. (Captura del sitio web de Instagram)	42
Figura 17. Visualización de posts de Instagram en la aplicación web. (Elaboración propia).....	44
Figura 18. Ejemplo perfil de Facebook. (Captura del sitio web de Facebook) ..	45
Figura 19. Ejemplo publicación de Facebook. (Captura del sitio web de Facebook).....	46
Figura 20. Visualización de perfil de Facebook en la aplicación web. (Elaboración propia).....	48
Figura 21. Ejemplo perfil de LinkedIn. (Captura del sitio web de LinkedIn)..	49
Figura 22. Visualización de perfil de LinkedIn en la aplicación web. (Elaboración propia).....	50
Figura 23. Visualización de perfil de LinkedIn en la aplicación web. (Elaboración propia).....	52

Figura 24. Diagrama de secuencia UML de la metodología de recolección de huella digital desarrollada. (Elaboración propia)	53
Figura 25. Escritorio de la distribución creada. (Elaboración propia).	54
Figura 26. Diagrama entidad relación de la base de datos. (Elaboración propia)	56
Figura 27. Diagrama de componentes de la herramienta. (Elaboración propia).	57

1 INTRODUCCIÓN

Desde el final de la Guerra Fría (Hassan & Hijazi, 2018) el desarrollo de Internet ha permitido que una gran cantidad de personas de distintas partes del mundo puedan comunicarse e intercambiar información digital llevándonos a lo que podríamos denominar como la era de la información. Aunque esta transformación de nuestro mundo ha traído consigo grandes beneficios para las sociedades actuales, también ha provocado el surgimiento de nuevos tipos de riesgos. Las nuevas tecnologías de comunicación e información son aprovechadas en la actualidad por la gran mayoría de organismos y empresas con el objetivo de mejorar sus servicios y del mismo modo también son utilizadas por cibercriminales, grupos terroristas, regímenes totalitarios y otro tipo de actores maliciosos para cometer sus crímenes.

Estos riesgos y necesidades llevaron a los gobiernos a investigar y desarrollar las herramientas y técnicas OSINT con el objetivo de hacerles frente y estar preparados para futuros nuevos riesgos o para poder ofrecer un mejor servicio a ciudadanos y clientes, entendiendo este término como toda la información disponible de forma pública.

Uno de los entornos en los que mayor cantidad de este tipo de información podemos encontrar, son las redes sociales. En la actualidad se han convertido en una forma indispensable de comunicación entre personas sobre todo en las nuevas generaciones. Es por esto por lo que la huella digital creada por una persona en las distintas cuentas de estas redes sociales cada vez que realiza una publicación o modifica alguno de sus datos presentados, puede ser de gran utilidad en un proceso de investigación sobre el individuo.

En este proyecto nos centraremos por lo tanto en las principales redes sociales utilizadas actualmente: Twitter, Instagram, Facebook y LinkedIn. Para ello tendremos que tener en cuenta la estructura de cada una de ellas para conocer los datos que podemos encontrar, como poder extraerlos y como identificar al individuo objetivo de nuestra investigación.

Finalmente, con la capacidad de realizar búsquedas utilizando el nombre y apellidos de una persona (además de datos adicionales que puedan acotar la búsqueda) para obtener la información publicada en sus redes sociales, permitirá facilitar la investigación a nivel temporal y operativo al contar con un entorno dedicado puramente al trabajo con OSINT.



2 ESTADO DE LA CUESTIÓN

Para poder entender cuál es el estado actual en cuanto a OSINT, tendremos que empezar con sus orígenes durante la Guerra Fría hasta llegar a la actualidad. Cada vez cobra más fuerza como campo de estudio para diversos sectores profesionales como pueden ser investigación criminal, inteligencia militar, realización de estudios sociológicos, etc.

2.1 ORIGEN OSINT

Resulta algo complicado situar el origen del término OSINT en el tiempo, pero tras la finalización de la Primera Guerra Mundial (Colquhoun, 2020) podemos encontrar registros sobre exmilitantes que decidieron dedicarse a recorrer el mundo entrevistando a importantes personalidades de los lugares a los que viajaba para escribir reportes que recibía el gobierno de los Estados Unidos. Estos reportes supondrían lo que más tarde pasaría a recibir el nombre de “inteligencia”, en este caso militar, y que hoy en día entendemos como el producto resultante de la información relativa a naciones extranjeras o fuerzas que puedan considerarse hostiles. El objetivo de esta inteligencia consiste en ayudar y guiar en la toma de decisiones estratégicas gracias al análisis de la información recopilada.

Tras el ataque de Pearl Harbour, la necesidad de inteligencia se hizo evidente llevando a Estados Unidos a crear la llamada OSS (*Office of Strategic Services*) que con el paso del tiempo pasaría a ser el precursor de El Departamento de Inteligencia de los Estados Unidos (INR) y la Agencia Central de Inteligencia (CIA). Este nuevo grupo se creó con la función de coordinar las actividades de espionaje llevadas a cabo por los Estados Unidos, además de llevar a cabo planes de propaganda y planificación posguerra entre otras tareas.

Dentro de la OSS se contaba con una rama dedicada a OSINT a pesar de ser un término desconocido por aquel entonces. Era la llamada *Research and Analysis Branch* que se encargaba de recopilar todo tipo de publicaciones en los medios de todo el mundo en busca de información que pudiera ser crucial para desarrollar inteligencia sobre los enemigos.

Tras el final de la Segunda Guerra Mundial no se produjeron grandes cambios en los sistemas de inteligencia OSINT comentados, pero durante la Guerra Fría (Schaurer & Störger, 2013) los países de ambos bandos ya contaban con equipos estables encargados de recopilar información de las fuentes abiertas que proporcionaban la mayor parte de la inteligencia utilizada.

Es durante este periodo de enfrentamiento, a finales de los años 80, cuando aparece por primera vez el término OSINT de la mano de El Ejército de Estados Unidos como respuesta a

la necesaria reforma de la inteligencia, dado la rápida evolución y cambios en los requisitos de la información. Fue a partir de entonces cuando los distintos organismos gubernamentales empezaron a trabajar para crear un entorno de trabajo en la inteligencia de fuentes abiertas como podrían ser el *Community Open Source Program Office (COSPO)* establecido en 1994 por la CIA o los manuales prácticos de trabajo publicados por la OTAN bajo el nombre de *NATO Open Source Intelligence Handbook*.

Con todo lo visto hasta ahora, hemos podido comprobar que el campo OSINT lleva tras de sí un largo recorrido que tiene como origen la necesidad de recabar información sobre los enemigos en los primeros conflictos bélicos de la era contemporánea, pero, como veremos en el siguiente apartado, dados los grandes cambios tecnológicos acontecidos en una fecha más reciente, la inteligencia en fuentes abiertas se ha modernizado y adaptado a las necesidades de nuestros días.

2.2 ESTADO OSINT EN LA ACTUALIDAD

El punto clave para entender el estado actual de la disciplina lo encontramos en la llamada Green Revolution del año 2009. Esta revolución tuvo lugar en Irán y fue motivada por el resultado de unas elecciones presidenciales que fueron consideradas fraudulentas.

Dada la fecha en la que se produjo esta revolución, Internet y las redes sociales ya se encontraban en funcionamiento (la red social Facebook surgió en el año 2004) con una actividad creciente y por primera vez, durante la Green Revolution, estos medios se inundaron de información civil sobre el evento político que acontecía. El uso de internet por parte de los ciudadanos iraníes aumento en un 14% (BBC NEWS, 2009) y en las redes sociales se podían ver noticias sobre lo que estaba ocurriendo en el país, así como campañas de protestas organizadas por los usuarios de la plataforma.

Con toda la información que podía encontrarse, cualquier persona de cualquier parte del mundo podía ser capaz de recopilar la información y escribir artículos o reportes que presentar como análisis de inteligencia sobre las revueltas iraníes. Por este motivo podríamos considerar este evento como el principal punto de inflexión con respecto a lo que es OSINT hoy en día.

En la actualidad, con el aumento de usuarios de Internet en miles de millones de personas, la información producida genera una huella digital de cada usuario inmensa y tanto las empresas como los organismos, han identificado la necesidad de invertir en el desarrollo de herramientas y técnicas de inteligencia. Es por ello que el campo de OSINT se encuentra a la orden del día.

Entre algunos de los principales y más claros usos que podemos encontrar en las noticias recientes de las tecnologías OSINT son los casos de lucha contra el ciberterrorismo. Y es que las redes sociales también son usadas por los grupos terroristas y otros tipos de entidades que pueden suponer un riesgo para los ciudadanos, como método de acción, comunicación y captación de nuevos miembros.

Las herramientas OSINT nos han permitido recabar la información pública que puede estar relacionada con este tipo de actividades criminales y analizarla consiguiendo la inteligencia necesaria para reducir el impacto negativo sobre nuestra sociedad.

Son muchos los ejemplos que pueden encontrarse sobre este uso de OSINT (Everstine, 2017) pero no es el único motivo por el cual esta rama está recibiendo tanta atención últimamente. La inteligencia que podemos generar a partir de la huella digital de una persona en Internet y sus redes sociales puede utilizarse en investigaciones de delitos, análisis de fraudes, estudios de reputación, investigación de currículos en procesos de selección, etc.

Con la finalización de este apartado podemos concluir que la investigación de fuentes abiertas está cobrando gran fuerza en la actualidad debido a la gran cantidad de información que podemos obtener a partir de la huella digital que los usuarios dejan en Internet. Como hemos visto, estos datos analizados correctamente nos permiten generar inteligencia que supone una gran ayuda a la hora de tomar decisiones en el mundo profesional.

2.2.1 TIPOS DE INFORMACIÓN DE FUENTES ABIERTAS

Es importante conocer qué tipo de información estamos tratando con el objetivo de distinguir con qué estamos trabajando en cada fase del proceso de obtención y análisis OSINT. Para ello, organizaciones como la OTAN (NATO, 2001) establecen en sus manuales clasificaciones de los datos que podemos encontrarnos al realizar una investigación OSINT. Son tres los términos clave que tenemos que diferenciar para entender las categorías en las que dividiremos la información de fuentes abiertas:

- a) Datos: Conjunto de hechos que describen algo sin ningún tipo de análisis o explicación
- b) Información: Datos que han sido interpretados con el objetivo de darles un uso significativo en el contexto en el que se trabaja.
- c) Conocimiento: Combinación de información y experiencia que ha sido aprendida tras la experimentación

Una vez conocemos estos términos, dividiremos los datos y la información que se obtiene de las fuentes abiertas en las siguientes categorías:

- a) *Open source data (OSD)*: Datos genéricos que provienen de una fuente primaria. Algunos ejemplos son fotografías, imágenes, grabaciones de audio o video, imágenes satélites.
- b) *Open source information (OSINF)*: Datos genéricos que han sido sometidos a algún tipo de filtro para cumplir con los requerimientos especificados. Son ejemplos de OSINF las búsquedas sobre un tema concreto o documentos orientados a un campo concreto (artículos, entrevistas...).
- c) *Open source intelligence (OSINT)*: Entenderemos como OSINT toda la información que haya sido recopilada, filtrada y designada a cumplir con un objetivo específico (resultado del procesado de los datos de fuentes abiertas).
- d) *Validated OSINT (OSINT-V)*: OSINT con un alto grado de certidumbre. Requiere de validación por parte de un tercero. Evita información falsa publicada en fuentes abiertas.

Cabe destacar, como se puede deducir de las definiciones anteriores, que los datos han de ser tratados para obtener información útil que pueda ser utilizada en el objetivo establecido para una investigación utilizando fuentes abiertas.

2.2.2 HERRAMIENTAS Y FRAMEWORKS OSINT

Tras haber visto algo de la historia de las tecnologías OSINT, debemos conocer cual el estado de las herramientas existentes para descubrir qué alternativas hay disponibles en la actualidad de cara a desarrollar una investigación en las fuentes abiertas.

Así pues, comenzaremos comentando algunas de las herramientas más utilizadas en el trabajo con OSINT destacando las siguientes:

- a) **Maltego (Maltego, 2021)**: Software utilizado para trabajo forense y OSINT enfocada en el descubrimiento de datos de fuentes abiertas permitiendo su visualización en formato gráfico.



Figura 1. Interfaz de Usuario tras realizar una búsqueda en Maltego. (Elaboración propia)

- b) **theHarvester (laramies, 2021):** Diseñada para ser utilizada en las primeras fases de un proceso de *pentesting*, utiliza las fuentes abiertas para determinar el entorno externo de una compañía en internet. Recopila información sobre emails, nombres, subdominios, IPs y URLs.

```

theHarvester
* TheHarvester Ver. 2.5
* Coded by Christian Martorella
* Edge-Security Research
* cmartorell@edge-security.com

Usage: theharvester options
-d: Domain to search or company name
-b: data source:
    google
    googleCSE
    bing
    bingapi
    pgp
    linkedin
    google-profiles
    people123
    jigsaw
    twitter
    googleplus
    all

-s: Start in result number X (default 0)
-v: Verify host name via dns resolution and search for virtual hosts
-f: Save the results into an HTML and XML file
-n: Perform a DNS reverse query on all ranges discovered
-c: Perform a DNS brute force for the domain name
-t: Perform a DNS TLD expansion discovery
-e: Use this DNS server
-l: Limit the number of results to work with (bing goes from 50 to 50 results)
-h: use SHODAN database to query discovered hosts
    google 100 to 100, and pgp doesn't use this option)

Examples:
theharvester -d microsoft.com -l 500 -b google
theharvester -d microsoft.com -b pgp
theharvester -d microsoft -l 200 -b linkedin
theharvester -d apple.com -b googleCSE -l 500 -s 300

```

Figura 2. Opciones de uso de theHarvester. (Elaboración propia)

- c) **Shodan (Shodan, 2021):** Motor de búsqueda que permite la búsqueda de dispositivos conectados a internet descubiertos mediante el uso de técnicas OSINT. Permite la búsqueda por servicio, localización, dispositivo y dominio entre otros.

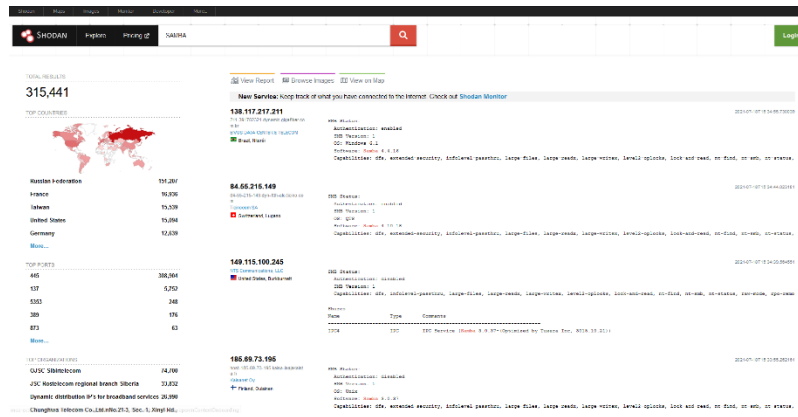


Figura 3. Resultados de búsqueda en Shodan. (Elaboración propia)

- d) **OSINT Framework (jnordine, 2021):** No es una herramienta para ser arrancada en una máquina o servidor, sino que nos ofrece una visión amplia de otros servicios o herramientas disponibles para la investigación de fuentes abiertas.

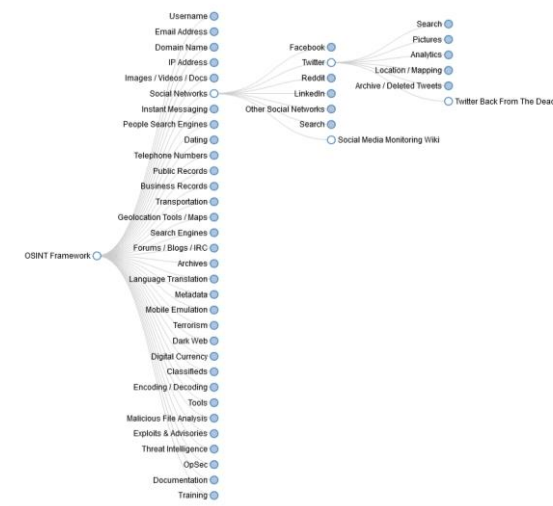


Figura 4. Sitio web de OSINT Framework. (jnordine, 2021)

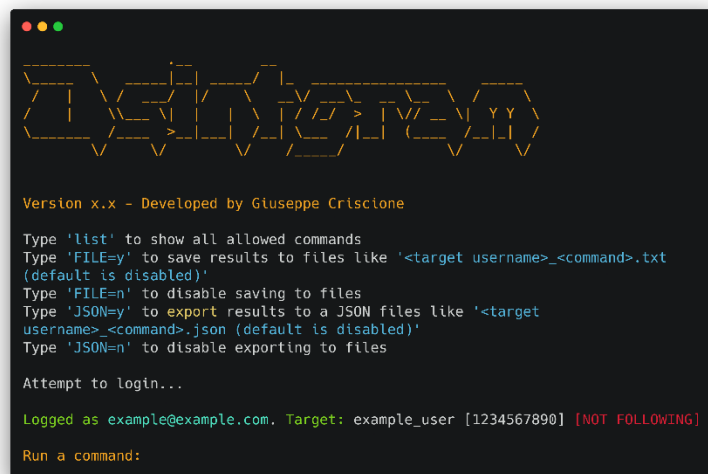
Si nos centramos en las redes sociales, encontramos herramientas dedicadas específicamente a este propósito entre las cuales vamos a destacar las descritas a continuación:

- a) **Sherlock (sherlock-project, 2021):** Herramienta orientada a la obtención de enlaces a las redes sociales de un usuario en función de un nombre de usuario. En algunas ocasiones se limita a concatenar el nombre de usuario a la URL base de la aplicación llevando a resultados incorrectos.

```
[+] Investing.com: Not Found!
[-] Issuu: Not Found!
[+] Itch.io: https://username.itch.io/
[-] Jindo: Not Found!
[-] Kaggle: Not Found!
[+] KanoWorld: https://api.kano.me/progress/user/username
[+] Keybase: https://keybase.io/username
[+] Kik: https://ws2.kik.com/user/username
[+] Kongregate: https://www.kongregate.com/accounts/username
[-] Launchpad: Not Found!
[+] LeetCode: https://leetcode.com/username
[-] Letterboxd: Not Found!
[+] LiveJournal: https://username.livejournal.com
[+] Mastodon: https://mstdn.io/@username
[+] Medium: https://medium.com/@username
[+] MeetMe: https://www.meetme.com/username
[+] MixCloud: https://www.mixcloud.com/username
[+] MyAnimeList: https://myanimelist.net/profile/username
[-] Myspace: Not Found!
[+] NameMC (Minecraft.net skins): https://namemc.com/profile/username
[+] Newgrounds: https://username.newgrounds.com
[+] OK: https://ok.ru/username
[+] Pastebin: https://pastebin.com/u/username
[+] Patreon: https://www.patreon.com/username
[-] Pexels: Not Found!
[+] Photobucket: https://photobucket.com/user/username/library
[+] Pinterest: https://www.pinterest.com/username/
[+] Pixabay: https://pixabay.com/en/users/username
[+] Plug.DJ: https://plug.dj/@/username
[+] Pokemon Showdown: https://pokemonshowdown.com/users/username
[+] ProductHunt: https://www.producthunt.com/@username
[-] Quora: Not Found!
[+] Rajce.net: https://username.rajce.idnes.cz/
```

Figura 5. Ejemplo de ejecución de Sherlock con el parámetro *username*. (sherlock-project, 2021)

- b) **Osintgram (Datalux, 2021):** Herramienta dedicada a la recolección, análisis y reconocimiento de perfiles de Instagram. Requiere del uso de una cuenta ya creada en la aplicación para su funcionamiento y es capaz de recopilar una gran parte de la información pública de un perfil, incluyendo números de teléfono de los seguidores de una cuenta.



```
Version x.x - Developed by Giuseppe Criscione

Type 'list' to show all allowed commands
Type 'FILE=y' to save results to files like '<target username>_<command>.txt'
(default is disabled)'
Type 'FILE=n' to disable saving to files
Type 'JSON=y' to export results to a JSON files like '<target
username>_<command>.json (default is disabled)'
Type 'JSON=n' to disable exporting to files

Attempt to login...

Logged as example@example.com. Target: example_user [1234567890] [NOT FOLLOWING]

Run a command:
```

Figura 6. Lanzamiento de herramienta Osintgram. (Datalux, 2021)

- c) **DumpltBlue (Le-tools, 2021):** Permite el volcado de contenidos de perfiles de Facebook de forma que se facilite su posterior análisis. Solo está disponible para la plataforma Windows y funciona como una extensión del navegador Google Chrome.

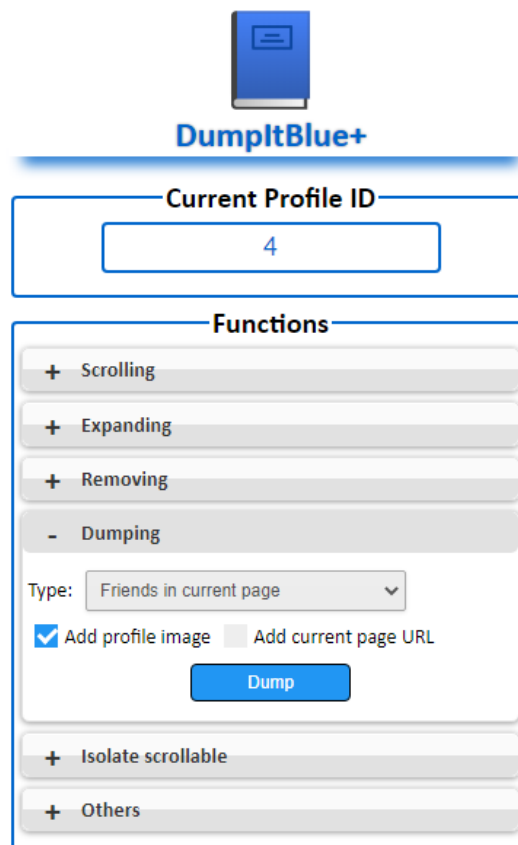


Figura 7. Panel de control de DumpItBlue+. (Le-tools, 2021)

Como hemos podido ver, existen una serie de herramientas OSINT que cuentan con un gran número de funcionalidades y entornos gráficos para el trabajo con fuentes abiertas en cualquier ámbito. Por otro lado, si buscamos algo específico para las redes sociales solo encontraremos pequeñas herramientas que van sufriendo las necesidades de cada red de forma independiente.

Algo parecido ocurre en cuanto a los entornos de trabajo o distribuciones orientadas al trabajo con este tipo de tecnologías ya que podemos encontrar algunas máquinas virtuales o distribuciones de sistema operativo orientadas a OSINT (aunque también suelen contener herramientas para análisis forense digital u otros campos), pero no encontramos resultados al buscar un *framework* orientado específicamente al trabajo con las redes sociales. Algunas de las distribuciones que podemos encontrar son: Trace Labs OSINT VM (Trace Labs OSINT VM, 2021) u OSINTUX (OSINTUX, 2018) siendo esta última una distribución Linux en castellano dedicada a inteligencia en fuentes abiertas.

Cabe destacar que otra de las herramientas que nos serán de gran utilidad en la investigación de fuentes abiertas serán los buscadores como Google ya que indexarán la información que podemos encontrar publicada en internet en función de las queries realizadas.

2.2.3 PUNTOS FUERTES DEL PROYECTO

Ahora que hemos centrado el contexto del proyecto podemos identificar cuáles son los aspectos más importantes del trabajo a desarrollar y sus puntos fuertes en función del estado actual de la materia.

El primero de los puntos fuertes que podríamos indicar sobre el trabajo sería el hecho de tratarse de un proyecto de desarrollo basado en un campo de la informática que está despertando un gran interés tanto en el mundo profesional como educativo por sus capacidades de aplicación hoy en día.

Continuando con esta idea, el segundo punto consistiría en la innovación aportada por el proyecto. Al tratarse de un tema relativamente reciente, el trabajo a realizar resulta distinto a lo ya creado y sirve como precedente para posibles futuros proyectos en relación a las redes sociales y OSINT. Además, la orientación del entorno de trabajo a las redes sociales supone un plus en este punto ya que son pocos los *frameworks* especificados en este aspecto de OSINT.

Por último, el proyecto en su conjunto establece una forma metodológica y ordenada de realizar investigaciones sobre las fuentes abiertas de información lo que estrecha el camino a la creación de estándares oficiales en materia de OSINT.

3 DESCRIPCIÓN DEL PROBLEMA

Las tecnologías y herramientas relacionadas con la inteligencia en fuentes abiertas son muchas y muy variadas dado el gran auge del campo en la actualidad. Sin embargo, existen dificultades a tener en cuenta de cara a empezar a trabajar en la materia debido a distintos factores.

En primer lugar, encontramos la dificultad de realizar una investigación OSINT que requiera el uso de herramientas totalmente distintas en cuanto a su uso, funcionalidad e implementación. Esto provoca que la curva de aprendizaje y el tiempo invertido en la investigación sean demasiado elevados para los recursos con los que se cuentan en algunas ocasiones.

Por otro lado, no existe una metodología estándar (existen metodologías, pero no son consideradas estándar) sobre los pasos a seguir para realizar una buena investigación OSINT. Los manuales y publicaciones existentes establecen principalmente el uso de algunas de las herramientas más utilizadas y definiciones de los principales términos sobre el tema, pero sería de gran utilidad contar con la existencia de procedimientos estándar específicos que permitieran a los investigadores seguirlos de forma inequívoca para alcanzar los objetivos buscados.

En el caso concreto de la huella digital de las redes sociales, los problemas anteriores son igualmente visibles dado que existen gran cantidad de herramientas que permiten la extracción de información de las cuentas de los usuarios utilizando técnicas de OSINT. Muchas de estas herramientas son de código abierto y en ocasiones son complejas de configurar o se han quedado obsoletas debido a los continuos cambios en la implementación de los sistemas y las políticas de acceso a las plataformas con el objetivo de evitar las prácticas de extracción de información. Pero el principal problema lo encontraríamos en la obtención de toda la información sobre un único individuo. Para conseguir este cometido se requiere conocer el nombre de usuario de la persona en cada una de las redes sociales que se quiere analizar algo que supone un problema ya que dependiendo de cada usuario las credenciales utilizadas pueden ser muy diversas y no tener ninguna relación con su identidad real.

Por último, son pocos los proyectos que podemos encontrar sobre distribuciones o *frameworks* de trabajo con conjuntos de herramientas OSINT y son aún menos comunes aquellos orientados a la huella digital y las redes sociales. La creación de este tipo de entornos de trabajo permitiría la portabilidad del sistema sobre el que se trabaja y facilitaría el trabajo de los investigadores al contar con todo lo necesario para trabajar con las fuentes abiertas en un único sistema.

En conclusión a este apartado, al tratarse de un tema relativamente reciente y en continua evolución, queda mucho trabajo que realizar de cara a formalizar y estandarizar el trabajo con las tecnologías OSINT y facilitar en gran medida el trabajo con este tipo de tecnologías y herramientas.

4 SOLUCIÓN PROPUESTA

Como solución al problema descrito en el apartado anterior, en el presente trabajo de fin de máster se desarrollará e implementará un entorno de trabajo OSINT que permita facilitar el trabajo con las fuentes abiertas y las redes sociales.

4.1 OBJETIVOS

El objetivo principal del trabajo consiste en la creación de una distribución de sistema operativo que contenga un conjunto de herramientas OSINT que faciliten trabajar con fuentes abiertas de cara al análisis de la huella digital en redes sociales. La distribución permitiría arrancarse con cualquier entorno de virtualización y comenzar a trabajar de manera rápida sobre fuentes abiertas sin realizar grandes procesos de configuración ni descargar herramientas adicionales. Esta distribución se publicará como un proyecto *Open Source* con el objetivo de que pueda ser utilizado por cualquiera y continuar con el desarrollo de la herramienta en función de sus necesidades que pueden ser compartidas a la comunidad.

Para ello, se tendrá que investigar cuales son las herramientas OSINT más utilizadas en la actualidad y preparar la imagen del sistema operativo más adecuada para el proyecto instalándolas y añadiendo la documentación necesaria.

Por otro lado, añadimos al proyecto otros dos objetivos que complementarán la distribución implementada. El primero de ellos consiste en el desarrollo de una aplicación web que permita la recolección de la huella digital de una persona en las principales redes sociales existentes. Las redes seleccionadas son Facebook, LinkedIn, Instagram y Twitter. Para cada una de estas redes será necesario comprender la estructura de la información que se presenta y cómo poder recopilar los datos utilizando técnicas OSINT. De igual manera, la herramienta permitirá la gestión de usuarios propios, para facilitar su despliegue en un entorno compartido en el caso de que el contexto lo requiera. También se permitirá realizar la búsqueda de una persona utilizando los datos personales que se posean de la misma para obtener los nombres de usuario utilizados en sus redes sociales en el caso de que se desconozcan.

La aplicación web deberá ser un entorno *“user friendly”* y contará con un manual de usuario para poder configurar el entorno a gusto del usuario. Las búsquedas realizadas por cada uno de los usuarios se almacenarán para mantener un historial de las consultas realizadas. Se

utilizará la tecnología de contenedores para permitir su despliegue de manera sencilla y portable.

Por último, en este documento se describirá el procedimiento seguido para realizar la recopilación de la información obtenida de las redes sociales a modo de metodología a seguir. El objetivo de este punto consiste en establecer una serie de pasos a seguir para extraer la huella digital de un usuario de las redes sociales y almacenarla de forma correcta para trabajar más tarde con ella.

4.2 LOGROS A ALCANZAR

Al tratarse de un trabajo basado en el desarrollo de herramientas de trabajo OSINT, definiremos los logros a alcanzar en función de los productos que se esperan obtener tras la realización del proyecto.

Estos productos consistirán en el contenedor de la herramienta OSINT desarrollada y la imagen de la distribución de sistema operativo creado con las herramientas (incluida la que vamos a desarrollar) instaladas.

De igual manera se buscará crear manuales de usuario para los dos productos comentados en el párrafo anterior. Estos manuales se publicarán en conjunto con las herramientas y esta memoria para solucionar cualquier problema o duda que pueda surgir a la hora de desplegar y utilizar todo el entorno desarrollado.

Por último, como logro a alcanzar con el desarrollo del trabajo fin de máster incluiremos conocer el estado actual en materia de OSINT relacionadas con la huella digital , así como el funcionamiento de las principales herramientas y usos de esta tecnología.

4.3 METODOLOGÍA

En cuanto a la metodología de trabajo que se llevará a cabo a lo largo del presente proyecto, estará basada en un desarrollo de tipo *bottom-up* basado en módulos. Esto consiste en comenzar implementando y probando pequeñas partes de la herramienta a desarrollar e ir conectando cada una de estas partes con el resto conforme se avanza en la ejecución del proyecto.

Comenzaremos por lo tanto estableciendo la estructura base de la herramienta web (*backend* y *frontend* básicos) para después implementar la recolección de datos de cada una de las redes sociales en módulos independientes que permitirán ser utilizados sin necesidad de la aplicación web, pero servirán como la lógica del *framework* una vez unidos entre sí.

Para completar todos los objetivos planteados, antes de comenzar con la implementación de la herramienta, será necesario realizar una investigación sobre los entornos que nos permitirán desarrollar el producto final que queremos generar. Igualmente, se requerirá de tiempo de estudio de las tecnologías desconocidas seleccionadas para adaptarse a ellas y conocer cuál es su funcionamiento.

Por último, una vez se haya desarrollado y preparado la herramienta web, se incluirá en la distribución previamente configurada para obtener la imagen final que constituirá el resultado del trabajo fin de máster.

Para el control de versiones y mantenimiento del código, se utilizará la herramienta GitHub donde se creará un repositorio privado que pasará a ser público una vez se haya concluido el proyecto.

4.4 PLANIFICACIÓN

La planificación de los distintos pasos a seguir para desarrollar el proyecto se ha planteado dividiendo cada una de las tareas a realizar y asignándoles una franja temporal en la que deben ser realizadas. Aunque las fechas de inicio y final de cada una de las tareas realizadas haya podido verse modificada a lo largo de la realización del proyecto, se ha representado la planificación utilizando un diagrama de Gantt para facilitar el seguimiento de cada una de las fases del desarrollo (Figura 1).

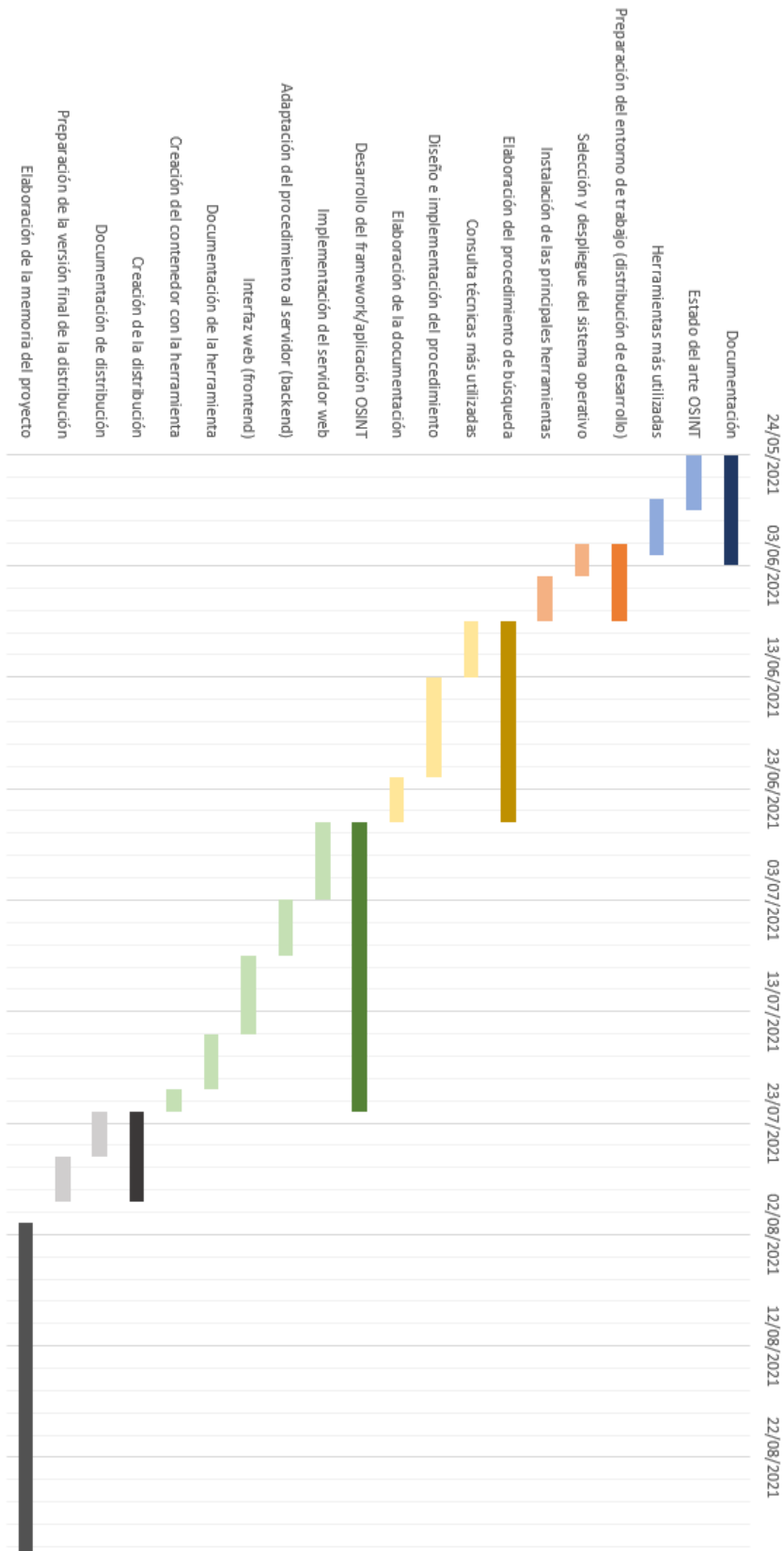


Figura 8. Diagrama de Gantt del proyecto



5 PRUEBAS Y VALIDACIÓN

5.1 ANÁLISIS ENTORNOS DE DESARROLLO APLICACIONES WEB

El primer paso, antes de comenzar con el desarrollo, consiste en la selección y análisis de los entornos de trabajo que se utilizarán para implementar las funcionalidades planteadas como objetivos. Para ello se realizó una búsqueda inicial para conocer cuáles son las alternativas más utilizadas en la actualidad a nivel profesional para desarrollar aplicaciones web.

En base a los conocimientos previos con los que se cuentan y la información recogida tras realizar la búsqueda, se concluyó en utilizar Django REST Framework (Christie, 2011) para la parte del servidor o *backend* y la biblioteca React de JavaScript (Facebook Inc., 2021) para la parte del cliente o *frontend*. Ambas tecnologías se utilizan actualmente en conjunto en gran cantidad de proyectos gracias a la capacidad de combinarlas de manera sencilla tal y como se puede ver en el esquema explicativo de la Figura 2.

En ambos casos, se utilizaron *templates* o *boilerplates* de proyectos ya creados que nos permiten contar con alguna funcionalidad esencial ya implementada y un sistema de ficheros del proyecto organizado correctamente.

A continuación, veremos el motivo de la elección de cada una de estas tecnologías, sus principales ventajas y los primeros pasos seguidos para comenzar a trabajar con ellas.

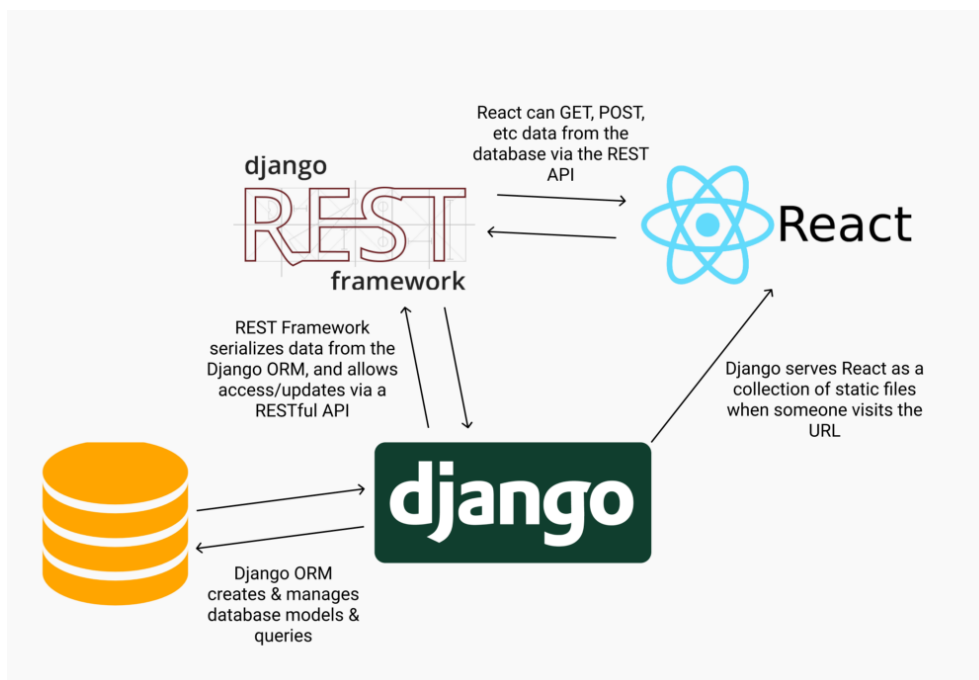


Figura 9. Tecnologías aplicación web. (Garner, 2021)

5.1.1 BACKEND: DJANGO REST API

El uso de APIs basadas en arquitectura REST (*Representational State Transfer*) conforma hoy en día el principal mecanismo de trabajo a la hora de crear aplicaciones web, debido a que constituye una forma estandarizada de intercambiar datos con otras aplicaciones de manera sencilla. La principal ventaja que nos ofrece la creación de una API REST, la encontramos por lo tanto en el establecimiento de una serie de llamadas en la parte del servidor para poder obtener los datos almacenados en las bases de datos desplegadas y poder mostrarlos en la parte del cliente.

Para la creación de este tipo de aplicaciones, existen distintas librerías o *frameworks* de los principales lenguajes de programación, que nos permiten simplificar el trabajo de manera significativa. En esta ocasión se seleccionó el Django REST Framework por los siguientes motivos:

- a) Framework basado en lenguaje Python, uno de los más utilizados en la actualidad.
- b) Permite la interacción con las bases de datos desplegadas utilizando Django ORM (*Object Relational Mapper*). Los datos almacenados en la base de datos se obtienen y crean sin la necesidad de realizar *queries* SQL. Permite la creación de los objetos de la base de datos mediante código Python.
- c) Genera API REST con documentación de forma automática.
- d) Muchas de las funcionalidades de una aplicación web vienen ya implementadas por defecto como podrían ser las estructuras y funciones de trabajo con usuarios de la herramienta.
- e) Experiencia con el entorno de proyectos ya ejecutados con éxito.

Como ya hemos comentado, para trabajar sobre un esquema de proyecto correcto, se utilizó un *boilerplate* de GitHub (Vivify-Ideas, 2021) basado en Docker. La principal ventaja que nos proporciona esta plantilla de trabajo, además de proporcionarnos un sistema de ficheros correctamente organizado, la encontramos en el uso de Docker para desplegar los elementos esenciales de la aplicación: base de datos PostgreSQL y servidor web Nginx. Aunque la plantilla escogida cuenta además con servicios de envío de correo y programación de tareas periódicas entre otras funcionalidades, para la realización de este proyecto solo se utilizaran los dos elementos comentados anteriormente.

Otra de las ventajas del *boilerplate* consiste en la gestión de usuarios de forma segura que viene implementada con la plantilla y nos ahorrará tener que gestionar esta tarea que resulta indispensable para la mayoría de las aplicaciones web.

5.1.2 FRONTEND: REACT COREUI

En cuanto a la parte del cliente, al igual que en el caso del *backend* se ha escogida una de las tecnologías predominantes en los desarrollos profesionales actuales como es React. Funciona de manera similar a Django en cuanto a que supone una interfaz de mayor nivel a la hora de crear interfaces de usuario interactivas utilizando el lenguaje JavaScript.

La experiencia con la que se contaba en cuanto a esta tecnología no era mucha por lo que ha requerido tiempo de aprendizaje, pero fue seleccionada debido a los siguientes factores:

- a) Tecnología flexible que puede ser utilizada con gran variedad de plataformas.
- b) Fácil integración con Django REST Framework.
- c) Es muy usada en la actualidad y cuenta con el soporte de Facebook.
- d) Amplia comunidad y soporte.
- e) Sencilla de usar y poca curva de aprendizaje.

Para la parte del cliente también se utilizó una plantilla de proyecto, en esta ocasión del *framework* llamado CoreUI (CoreUI, 2021) que nos permitirá contar con una base para el *frontend* de nuestra aplicación que podremos modificar a nuestro gusto para conseguir los resultados buscados.

La plantilla no estaba basada en Docker en esta ocasión, sino que se instalaba directamente utilizando la funcionalidad npm, pero con pequeños cambios sobre el *dockerfile* encargado de desplegar la parte del servidor, se consiguió levantar ambas partes de manera simultánea.

5.2 DESARROLLO APLICACIÓN WEB

Como se comentó en el apartado de metodología, el desarrollo de la aplicación se llevará a cabo creando módulos independientes para cada una de las redes sociales, pero antes de eso, será necesario preparar el entorno de trabajo con las tecnologías seleccionadas.

A lo largo de este punto veremos cómo se preparó el entorno inicial en base al cual se ha desarrollado el proyecto utilizando los *templates* de los *frameworks* escogidos. Tras esto se comentará la implementación de cada uno de los módulos atendiendo a las peculiaridades y problemas que han presentado cada una de las redes sociales.

5.2.1 PREPARACIÓN DEL ENTORNO DE TRABAJO

Tras comprender el funcionamiento de los *frameworks* y plantillas de Django y CoreUI, el siguiente paso consiste en probar ambos entornos en conjunto. Para ello se modificó el archivo utilizado por docker-compose para levantar cada uno de los servicios, añadiendo la

parte del cliente y estableciendo el mapeado de puertos entre los contenedores y la máquina donde se ejecutan.

Con esta parte funcionando, comprobamos que la aplicación se encuentra levantada con las funcionalidades básicas implementadas en las plantillas como el *login* y registro de usuarios que se puede ver en la Figura 3.

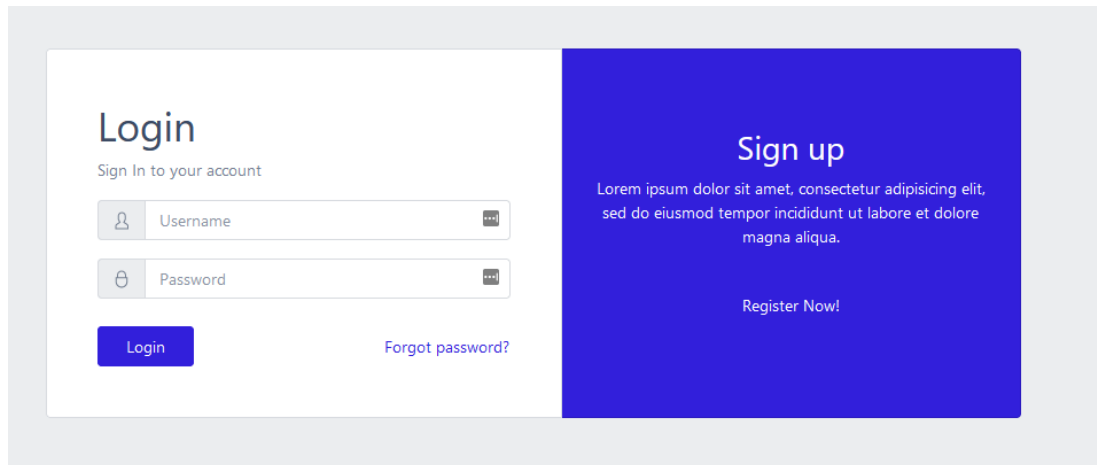


Figura 10. Ventana de Login de la aplicación. (Elaboración propia)

La gestión de usuarios llevada a cabo en la parte del *backend* está implementada para que funcione de forma segura utilizando cifrado en las credenciales guardadas en la base de datos y autenticación mediante la tecnología JWT (JSON Web Token).

El *framework* de Django nos ofrece la posibilidad de visualizar una API interactiva que facilitará en gran medida el trabajo a la hora de implementar los módulos de las redes sociales. Esta API se genera gracias a la librería Swagger y nos será de gran utilidad para comprobar que la parte del servidor funciona correctamente antes de integrarla en conjunto con el cliente. Para acceder a ella simplemente tendremos que crear un primer usuario nuevo en la aplicación accediendo al contenedor de Docker que está ejecutando la aplicación y buscar en el navegador la ruta en la que se nos mostrara la API interactiva tal y como se ve en la Figura 4.

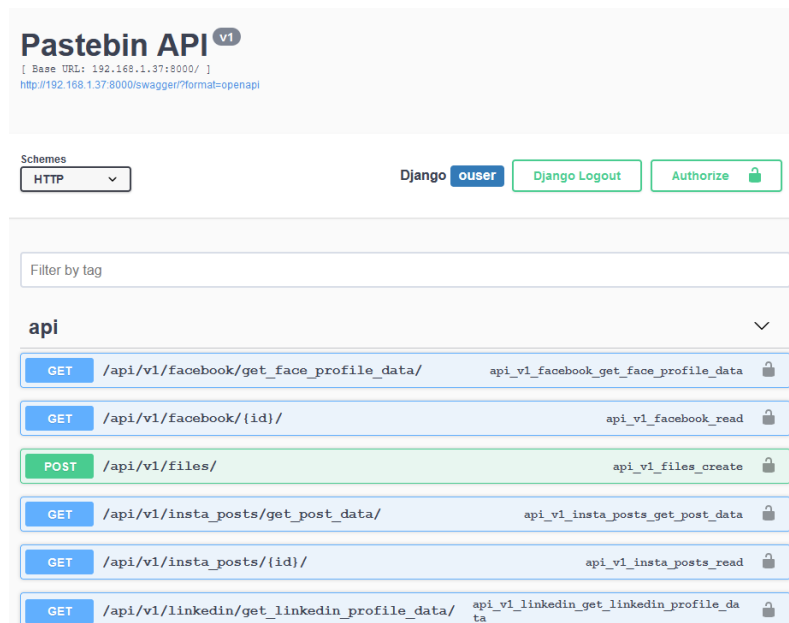


Figura 11. API interactiva de Swagger. (Elaboración propia)

5.2.2 TWITTER

El primer módulo a implementar fue el módulo de la red social Twitter. Lo primero que debemos tener en cuenta en esta ocasión es el formato de las publicaciones de Twitter y la información que podemos obtener del perfil de cada usuario.

El perfil de un usuario de Twitter cuenta con una biografía sobre el usuario y fotografías de portada y de perfil principalmente, como puede verse en la figura 5. También será información importante sobre el usuario, elementos como el número de seguidores y personas seguidas ya que nos pueden ayudar a identificar la persona que estamos buscando.



Figura 12. Ejemplo perfil de Twitter. (Captura del sitio web de Twitter)

Pero la parte importante de la red social la encontramos en los posts o tweets donde los usuarios publican texto de hasta un total de 280 caracteres que puede venir acompañado de imágenes, videos o enlaces a otras publicaciones (Figura 6).



Figura 13. Ejemplo de tweet. (Captura del sitio web de Twitter)

Tras haber analizado que información podemos recopilar de una publicación de la red social, el siguiente paso será buscar la herramienta que nos permita extraer los datos. Al tratarse de datos presentados en un sitio web, se utilizarán los llamados *scrapers* que nos permiten extraer la información de las distintas páginas web mostradas por un navegador de forma automatizada.

Son muchos los *scrapers* que podemos encontrar para la red social Twitter pero en este caso utilizaremos la herramienta Twint ya que puede utilizarse como una librería de Python y es una de las que cuenta con mayores funcionalidades. Twint cuenta con distintos comandos que nos permiten recolectar los tweets de un determinado usuario sin la necesidad de utilizar una cuenta ya creada de Twitter para realizar las búsquedas (siempre y cuando el usuario buscado no tenga la cuenta privada, en este caso necesitaríamos una cuenta que fuera un seguidor admitido).

Para implementar este módulo se ha creado un *script* en el lenguaje Python que utilizando la librería Twint, recopila toda la información del usuario con el *username* indicado y filtra los resultados recibidos para crear el objeto que se guardará en la base de datos en función de lo que se haya encontrado. En concreto, se creará una lista de diccionarios en la que cada diccionario constituirá la representación de un tweet del usuario. Para el caso concreto del contenido multimedia que se publica en la red, contamos con la ventaja de que al recoger la

información de un Tweet también recuperamos la URL de los elementos multimedia y esta URL es accesible desde nuestra aplicación por lo que solo tendremos que realizar una petición a esta URL para mostrar las imágenes en nuestro cliente.

Ahora que ya conocemos la información que podemos obtener y su estructura, el siguiente paso será crear los modelos de la base de datos para poder almacenar los datos. Dividiremos por lo tanto la información de perfil de la de un tweet ya que esto nos permitirá contar con una tabla Perfil en la base de datos que recoja los datos de los perfiles de las distintas redes que un usuario posee y, por otro lado, una tabla Tweets donde se almacenen las publicaciones en la red social. Las estructuras de datos a utilizar en este caso pueden verse en las Tablas 1 y 2.

Dato	Tipo de dato	Descripción
<u>id</u>	Entero	Identificador único de los perfiles de un individuo.
twitter_profile_image	Texto	Cadena de texto que indica la url donde se localiza la imagen de perfil.
twitter_bio	Texto	Biografía de twitter del usuario.
twitter_followers	Entero	Número de seguidores de Twitter.
twitter_followed	Entero	Número de seguidos en Twitter.

Tabla 1. Modelo base de datos de perfil de Twitter

Dato	Tipo de dato	Descripción
<u>Id</u>	Entero	Identificador único de un tweet.
Username	Texto	Nombre de usuario de Twitter.
tweet_data	Texto	Texto contenido en el tweet.
profile_name	Entero	Nombre de perfil del usuario. Distinto del nombre de usuario.
Link	Texto	Enlace al tweet.
Datetime	Fecha	Fecha de publicación del tweet.
tweet_image	Texto	Cadena de texto que contiene la url donde se localiza la imagen adjuntada a un tweet. En

		el caso de que no se haya publicado ninguna imagen permanecerá con el valor nulo.
--	--	---

Tabla 2. Modelo base de datos de publicación de Twitter.

Tras crear los modelos de la base de datos y toda la lógica a nivel de *backend*, comprobamos que todo funciona utilizando la API REST generada por Django. Este proceso se repetirá para cada una de las redes sociales con el objetivo de asegurar que la integración con el cliente responde adecuadamente.

En cuanto a la parte del cliente, crearemos la función que se encarga de realizar la llamada a la API utilizando la librería Axios para recoger los tweets y la importamos en el componente del *framework* React que configura la interfaz que recibirá el navegador. El resultado final puede verse en la Figura 7.

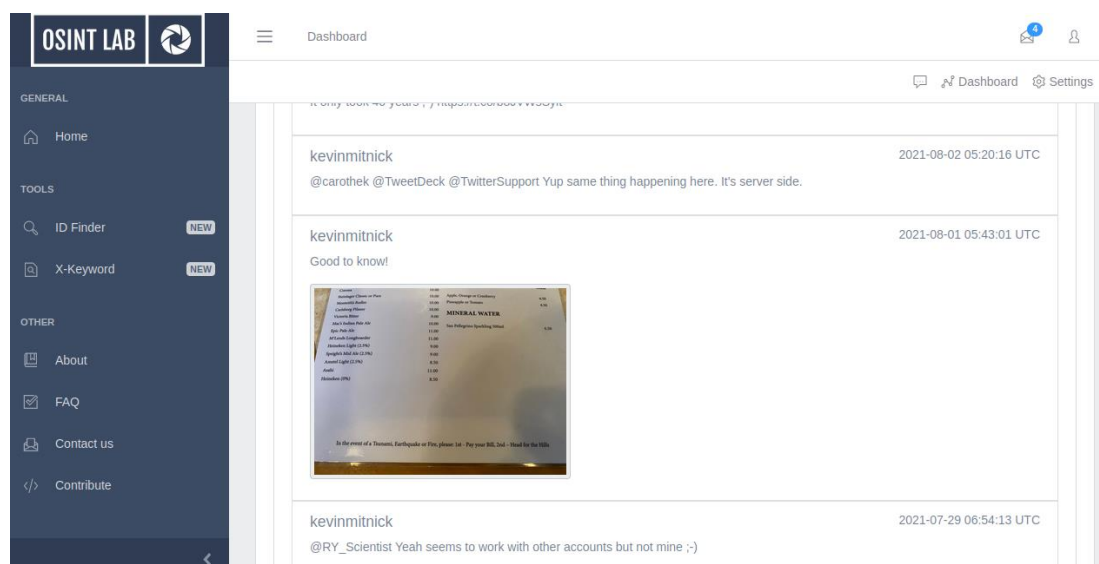


Figura 14. Visualización de tweets en la aplicación web. (Elaboración propia)

5.2.3 INSTAGRAM

El siguiente módulo por implementar será el de la red social Instagram y con tan solo un primer vistazo a los perfiles de esta, ya podemos ver que cuenta con algunas peculiaridades que la diferencian del resto.

Lo primero que llamaría la atención a un nuevo usuario es que las publicaciones de la red están orientadas al contenido audiovisual ya que todo *post* viene acompañado de una imagen o video y el texto de descripción queda relegado a un segundo plano. Por este motivo nos tendremos que centrar en la recolección de este contenido multimedia que como veremos más adelante, no será tan sencillo como en el caso anterior. La red social también

permite añadir contenido temporal que permanece visible a los usuarios durante 24 horas y pasado este periodo temporal desaparece. Son las llamadas *Stories* que en este caso obviaremos ya que se planteara como trabajo futuro.

Analizando un perfil cualquiera (Figura 8) podemos ver que la información general es común a la de Twitter. Cada perfil cuenta con una imagen de perfil y una pequeña biografía sobre el usuario. Además, cada cuenta tiene un número de seguidores y de seguidos que también nos aportarán información sobre los usuarios.

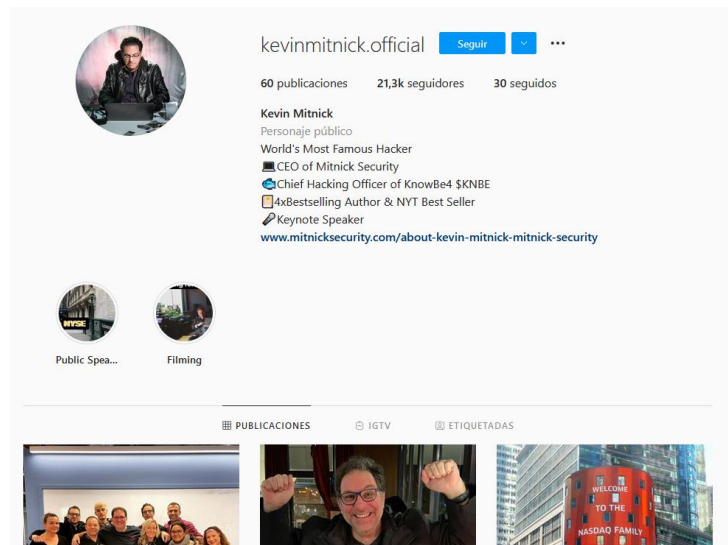


Figura 15. Ejemplo perfil de Instagram. (Captura del sitio web de Instagram)

Si a continuación consultamos una de las publicaciones veremos que la información que podemos extraer de ella, además de la propia imagen, será el texto descriptivo y el número de “me gusta” o “likes” recibidos (Figura 9).



Figura 16. Ejemplo publicación de Instagram. (Captura del sitio web de Instagram)

Tras conocer la estructura de las publicaciones y los perfiles de una red social, procederemos de igual manera que en el caso anterior, buscando el *scraper* que mejor se adapte a nuestras necesidades.

Aquí encontraremos uno de los primeros problemas y es que la mayoría de las herramientas OSINT que podemos encontrar para extraer información de esta red social, descargan las publicaciones a un directorio local pero no nos permiten integrarlas como una librería de algún lenguaje de programación en nuestro código. Además, resulta complicado extraer el resto de información de una publicación haciendo uso de ellas ya que la documentación no suele indicar que estructuras de datos utilizan y en muchas ocasiones la descargan como un fichero de texto.

Sin embargo, tras buscar alguna herramienta que estuviera desarrollada en Python y que por lo tanto pudiera ser importada como una librería en un *script* se encontró Instaloader (Instaloader, 2021). Al igual que el caso de Twint, nos permite descargar las publicaciones de un usuario sin necesidad de hacer *login* siempre y cuando la cuenta sea pública.

Crearemos un *script* de Python que nos permita descargar las fotos y crearemos una nueva lista de diccionarios que contendrá los datos de un post en cada uno de los diccionarios. Al igual que en Twitter, las imágenes de cada *post* se recopilan almacenando la URL a la que hace referencia, pero en este caso, Instagram no permite recuperar sus imágenes desde otro sitio web. La solución a este problema se tomo descargando la imagen en local utilizando la librería *requests* de Python e indicando la URL del recurso local en el objeto de la base de datos. Aunque puede parecer que esto lo podíamos haber hecho con cualquier otra herramienta OSINT de Instagram como hemos comentado al inicio de este apartado, cabe destacar que, al estar trabajando con un *script* propio de Python, la gestión de los objetos a la hora de crear los modelos de la base de datos y la integración con la API son mucho más sencillas.

Tras tener comprobar el funcionamiento del *script* creado, implementamos un nuevo módulo Django que nos permita configurar la nueva parte de la API y creamos los modelos de la base de datos que contendrán la información que puede verse en las Tablas 3 y 4.

Dato	Tipo de dato	Descripción
<u>id</u>	Entero	Identificador único de los perfiles de un individuo.
insta_profile_image	Texto	Cadena de texto que indica la url donde se localiza la imagen de perfil.
insta_bio	Texto	Biografía de Instagram del usuario.

insta_followers	Entero	Número de seguidores de Instagram.
insta_followed	Entero	Número de seguidos en Instagram.

Tabla 3. Modelo base de datos de perfil de Instagram

Dato	Tipo de dato	Descripción
<u>id</u>	Entero	Identificador único de un post de Instagram.
username	Texto	Nombre de usuario de Instagram.
post_data	Texto	Texto contenido en el post.
datetime	Fecha	Fecha de publicación del post.
post_image	Texto	Cadena de texto que contiene la url donde se localiza la imagen de un post.
likes	Entero	Número de <i>likes</i> de la publicación

Tabla 4. Modelo base de datos de publicación de Instagram.

Desde el cliente solo tendremos que crear una nueva función axios que nos devuelva el objeto con la lista de *posts* del usuario buscado, y mostrar las imágenes accediendo a la URL local tal y como hemos explicado. Tras integrarlo todo y desplegar el entorno con los nuevos cambios el resultado puede verse en la imagen inferior (Figura 10).

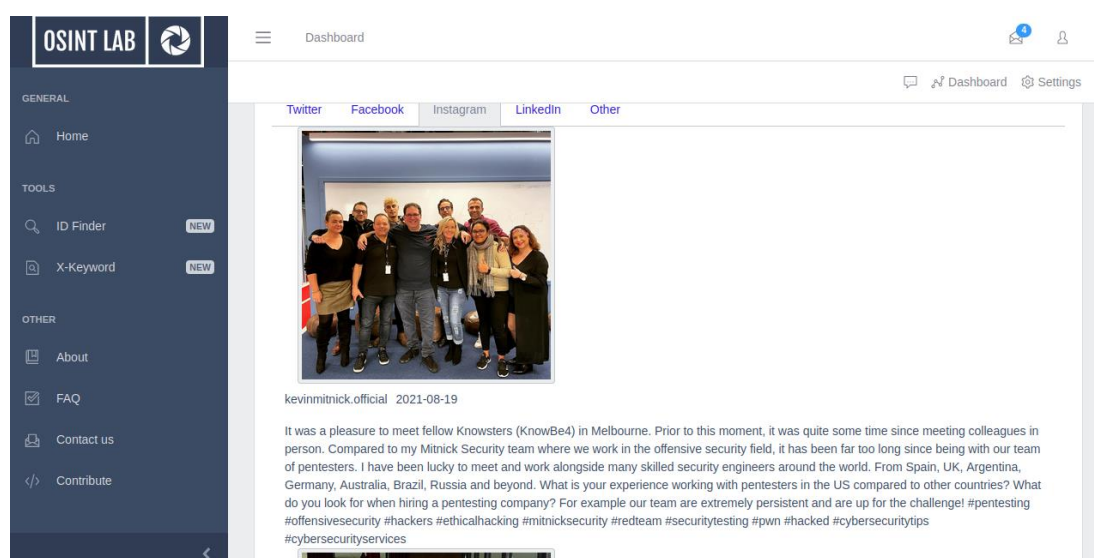


Figura 17. Visualización de posts de Instagram en la aplicación web. (Elaboración propia)

5.2.4 FACEBOOK

Pasando a la penúltima red social que trataremos en este proyecto, nos encontramos con Facebook, una de las principales y más usadas hasta la fecha. Procediendo del mismo modo que en los casos anteriores, pasamos a analizar el perfil de nuestro amigo Kevin Mitnick para conocer cual es la estructura de la plataforma.

Tal y como podemos ver en la Figura 11, la principal información que podemos obtener de un perfil consta de información personal donde podemos incluso encontrar un número de teléfono adjuntado por el usuario entre otros métodos de contacto. Además, podemos ver la función profesional que desempeña, así como sus publicaciones a la venta y los eventos que tiene planeados en un futuro.



Figura 18. Ejemplo perfil de Facebook. (Captura del sitio web de Facebook)

Las publicaciones tienen un aspecto muy similar a las de la red social Instagram con la única diferencia de que en este caso podemos encontrar alguna de ellas en las que no se adjunta ningún tipo de contenido multimedia. En esta ocasión, otra peculiaridad que encontramos es que las recciones pueden ser de distinta naturaleza y están basadas en “emoticonos”. En la Figura 12 puede verse un ejemplo de publicación.



Figura 19. Ejemplo publicación de Facebook. (Captura del sitio web de Facebook)

Aunque la estructura de la red social no parece que vaya a suponer un inconveniente para la implementación del módulo de recolección de datos, al realizar la búsqueda de herramientas OSINT o *scrapers* para Facebook, encontramos que hay una gran escasez para esta red social. Algunas de las herramientas que se probaron no funcionaban o daban problemas a la hora de intentar integrarlas con la nuestra aplicación. Esto es debido a que la red social cuenta con políticas muy restrictivas en cuanto a la obtención de datos de su plataforma, lo cual directamente nos advierten los desarrolladores de herramientas que nos alertan de las consideraciones para tener en cuenta antes de utilizar esta técnica sobre la red social (Octoparse, 2020).

La solución que se decidió tomar fue utilizar la herramienta *Facebook_scrapper* que permite ser integrada en un *script* de Python y permite recopilar únicamente la información del perfil. Esto es debido a que la herramienta se aprovecha del sitio web móvil de Facebook para obtener los datos sobre un determinado usuario y en esta plataforma, la información que se puede obtener en estos dispositivos está técnicamente desorganizada de tal manera que al recoger los *posts* sobre un determinado usuario los resultados que aparecen en muchos

casos no tienen relación con el investigado, sino que son menciones por parte de terceros que tampoco tienen relación alguna con el propietario del perfil.

Otra de los puntos que se debieron tener en cuenta en este caso es que la herramienta no funciona si no se le proporciona una *cookie* de sesión de un usuario válido por lo que se creo una cuenta falsa sin proporcionar información personal real para poder obtener la *cookie* una vez realizado el *login*.

Tras haber implementa el *script* y comprobado su funcionamiento, se comprobaron que datos se podía recopilar utilizando *Facebook_scrapper* de cara a crear los modelos de la base de datos con la mayor información posible. El resultado obtenido puede verse en la Tabla 5.

Dato	Tipo de dato	Descripción
<u>id</u>	Entero	Identificador único de los perfiles de un individuo.
name	Texto	Nombre personal del usuario de la cuenta.
username	Texto	Nombre de usuario del perfil.
category	Texto	Categoría seleccionada por el usuario de la cuenta. Utilizada en empresas y personalidades famosas.
profile_picture	Texto	Cadena de texto que indica donde se localiza la imagen de perfil.
basic_info	Texto	Información básica sobre el usuario.
contact_info	Texto	Información de contacto proporcionada por el usuario. Puede incluir números de teléfono o direcciones de correo electrónico.
places_lived	Texto	Lugares de residencia del usuario.
education	Texto	Lugares en los que ha estudiado el usuario.
family_members	Texto	Otros usuarios de Facebook con los que se mantiene algún tipo de relación familiar.

life_events	Texto	Eventos planeados por parte del usuario.
favourite_quotes	Texto	Frases o citas favoritas del usuario.

Tabla 5. Modelo base de datos de perfil de Facebook

Con los modelos listos y la API probada, la integración con el *frontend* se lleva a cabo de igual manera que en los casos anteriores y los resultados finales pueden verse en la Figura 13.

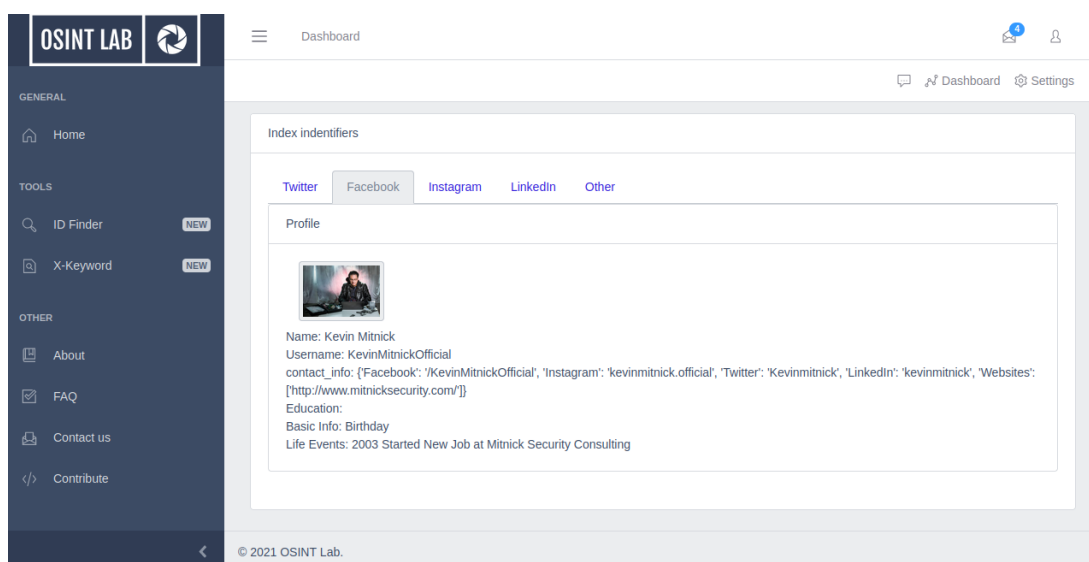


Figura 20. Visualización de perfil de Facebook en la aplicación web. (Elaboración propia)

5.2.5 LINKEDIN

Llegando a la última parte de este apartado vamos a hablar sobre la red social LinkedIn. Esta plataforma tiene como público objetivo el sector profesional de tal manera que podemos encontrar perfiles de empresas o perfiles profesionales de usuarios. En esta ocasión dado que el alcance del trabajo está orientado hacia la huella digital, nos centraremos únicamente en la información que podemos extraer de los usuarios.

En cuanto a la información que podemos obtener de un perfil como el de la Figura 14, vemos que lo primero que encontramos son datos sobre la ocupación profesional del usuario e información de contacto. Como en el resto de las redes, la página principal de un usuario también cuenta con un apartado a modo de biografía, pero en esta ocasión LinkedIn enfoca el resto de sus apartados a presentar los logros profesionales o aptitudes del usuario con la intención de presentar sus cualidades a las empresas que se encuentran en un proceso de selección.

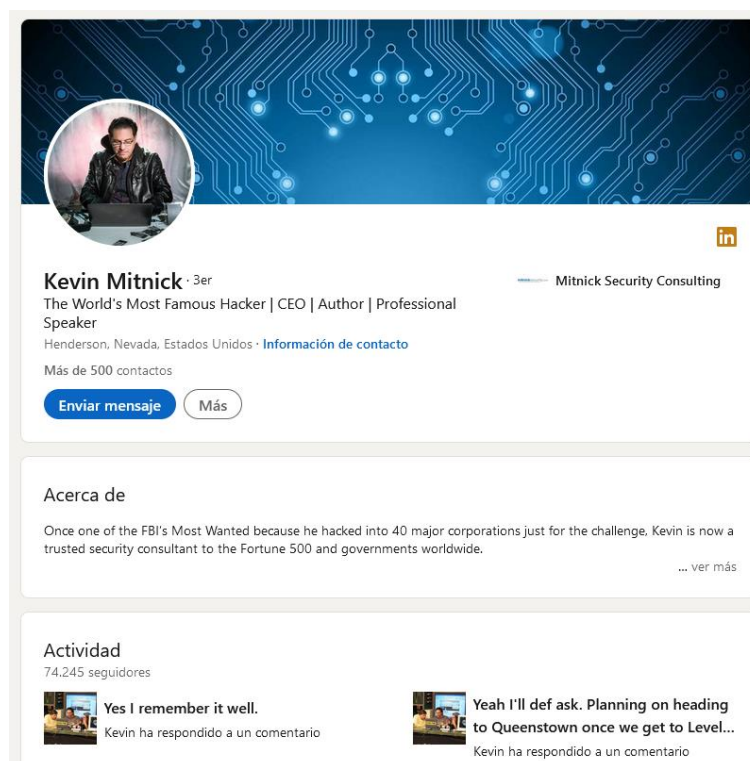


Figura 21. Ejemplo perfil de LinkedIn. (Captura del sitio web de LinkedIn)

Aunque LinkedIn cuenta también con un sistema de publicaciones y anuncios, la mayor parte de la información sobre un usuario, ya que es el núcleo del funcionamiento de la red, lo encontramos en los perfiles donde nos centraremos para la extracción de la huella digital.

Con LinkedIn ocurre algo similar a Facebook a la hora de recopilar información de la plataforma ya que las políticas son restrictivas (menos que Facebook) y las herramientas que existen son pocas y con errores en algunos casos. La herramienta escogida para este módulo es *linkedin-scraper* una librería de Python que nos permite recopilar información sobre los perfiles de usuarios de la red. La principal peculiaridad de este *scraper* consiste en que será necesario utilizar un componente extra para hacerla funcionar, una instancia de Selenium. Selenium utiliza los navegadores convencionales en conjunto con *web drivers* para poder automatizar el proceso de búsqueda que se realizaría a través de programas como Chrome, Firefox, Opera, etc. Esto nos obliga a tener instalado alguno de los navegadores más usados para poder simular la navegación, pero en esta ocasión, dado que estamos trabajando con un entorno basado en Docker, añadiremos un nuevo contenedor que se encargara de ejecutar Selenium y desde el servidor que corre el *backend* atacaremos de forma remota al nuevo *container*.

Tras comprobar la información que se pueden extraer ejecutando la herramienta el modelo de la base de datos corresponde al de la Tabla 6.

Dato	Tipo de dato	Descripción
<u>id</u>	Entero	Identificador único de los perfiles de un individuo.
name	Texto	Nombre personal del usuario de la cuenta.
profile_link	Texto	Enlace a la cuenta de LinkedIn del usuario.
about	Texto	Información a modo de biografía sobre el usuario.
experiences	Texto	Conjunto de experiencia profesional del propietario de la cuenta.
educations	Texto	Historial de estudios del propietario de la cuenta.
accomplishments	Texto	Logros del propietario de la cuenta.
interests	Texto	Intereses del propietario de la cuenta.
contacts	Texto	Información de contacto del usuario de la cuenta.

Tabla 6. Modelo base de datos de perfil de LinkedIn

La integración con el cliente se realiza de igual manera que en los casos anteriores. Tras comprobar el correcto funcionamiento de la API se crean las funciones axios y se realizan las correspondientes llamadas en las interfaces de React.

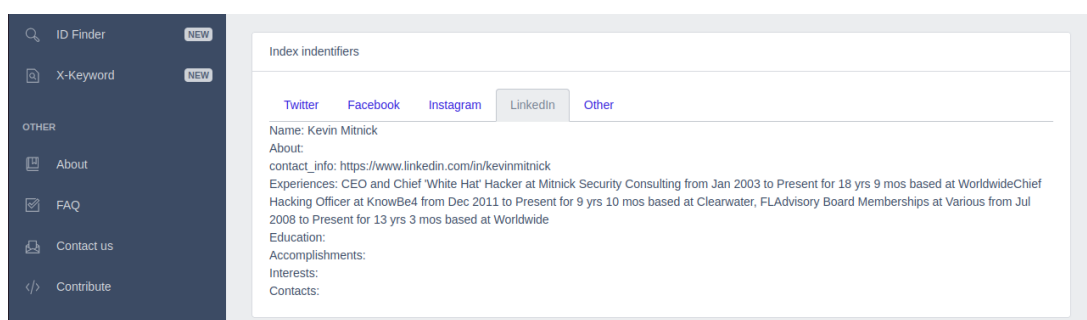


Figura 22. Visualización de perfil de LinkedIn en la aplicación web. (Elaboración propia)

5.2.6 BÚSQUEDA DE PERSONAS Y METODOLOGÍA

Ahora que ya tenemos todos los módulos implementados, vamos a elaborar una metodología que nos permita obtener los nombres de usuario de una persona a partir de sus datos personales. Para ello, la mejor de las alternativas que podemos utilizar serán los buscadores convencionales y en esta ocasión nos centraremos en el más usado de ellos Google. La herramienta cuenta con operadores booleanos que permiten reducir el campo de búsqueda que para el caso de las redes sociales se pueden utilizar con el símbolo @ seguido del nombre de la plataforma en la que queremos buscar el usuario (ej: Kevin Mitnick @facebook).

Tras realizar pruebas utilizando este mecanismo, se pudo comprobar que los resultados obtenidos eran completamente distintos a los esperados en el caso de que no se tratara de una personalidad famosa llegando a obtener links que no tienen ninguna relación con los sitios web oficiales de las redes sociales. Eliminando el operador booleano @ los resultados obtenidos son mucho mejores ya que todos los resultados devueltos por Google pertenecen al dominio de las redes sociales lo que nos asegura que no accedemos a ningún sitio web que pueda ser peligroso.

La mejor manera de realizar la búsqueda será por lo tanto añadiendo la red social de la cual queramos buscar el nombre de usuario del individuo investigado y analizar los resultados obtenidos. En la mayoría de los casos, para localizar a personas que no cuentan con gran interacción en las redes, los resultados no serán correctos y tendremos que añadir otros datos que conozcamos como pueden ser ciudad de residencia, profesión, empresa en la que trabaja, etc.

En nuestro caso utilizaremos únicamente el nombre del usuario con el objetivo de agilizar las búsquedas y para obtener los resultados de las búsquedas se utilizará la librería Google de Python (Googlesearch, 2018). Con la ayuda de esta librería implementaremos un *script* que realice las búsquedas sobre Google obteniendo la URL del perfil que aparezca como primer resultado, pero para poder recolectar posteriormente la huella digital de cada una de las redes, necesitaremos obtener el *username* del usuario.

Si analizamos las URLs obtenidas, podemos ver que todas ellas cuentan con una estructura establecida a la hora de identificar los perfiles del usuario por lo que utilizaremos esto a nuestro favor para extraer los nombres de usuario. Tanto Twitter, como Instagram y Facebook, concatenan el nombre de usuario al dominio de la red social de tal manera que, si separamos la cadena de la URL obtenida en función del operador '/', obtendremos el nombre

de usuario que más tarde podremos utilizar para recopilar los datos del usuario utilizando los módulos creados anteriormente. Para LinkedIn, la única diferencia que encontramos consiste en que además del dominio de la aplicación, también se añade la cadena '/in' por lo que la separación será distinta ya que habrá que omitir esta parte.

Como ya se ha comentado, en muchos casos los resultados utilizando únicamente el nombre personal del usuario, pueden no ser correctos. En estos casos la investigación requiere de mayor conocimiento sobre la persona y la búsqueda puede llegar a ser muy específica en función de la persona a investigar. Por esto, la decisión tomada en el desarrollo de la aplicación web es permitir introducir los nombres de usuario de cada una de las redes en el caso de que ya se conozcan o ya hayan sido descubiertos por el investigador de manera independiente.

Tal y como se ve en la Figura 16, el cliente mostrará un campo de búsqueda en el que introducir el nombre personal del individuo y cuatro campos en los que se podrá introducir el nombre de usuario de la red social. Si los campos de las redes sociales no se rellenan y la *check box* de esa red está marcada, se realizará la recolección de huella digital utilizando la búsqueda mediante Google implementada.

The screenshot shows the OSINT LAB web application interface. On the left is a dark sidebar with navigation links: Home, ID Finder, X-Keyword, About, FAQ, Contact us, and Contribute. The main content area has a 'Search Filters' section with a 'Name' input field containing 'kevin mitnick', an 'Estimated Results' count of 100, and checkboxes for social media filters (Twitter, Facebook, Instagram, LinkedIn). Below this are input fields for usernames for each network. A 'Recent Searches' table is also visible, showing search history with columns for Name, Date, and Social Media.

Name	Date	Social Media
Cristiano Ronaldo	2021-10-05 18:00:02 UTC	Twitter, Instagram
Chema Alonso	2021-10-05 18:01:10 UTC	Twitter, Instagram, LinkedIn

Figura 23. Visualización de perfil de LinkedIn en la aplicación web. (Elaboración propia)

Con todo lo que hemos visto a lo largo de los anteriores apartados, podemos establecer los pasos a seguir para obtener la huella digital de una persona a partir de algunos de sus datos personales. En primer lugar, necesitaremos hacer uso de los buscadores web convencionales para localizar el sitio web que nos muestra el perfil del usuario en cada una de las redes

sociales. Tras esto, el siguiente paso será localizar el nombre de usuario que utiliza la plataforma para identificarlo en conjunto con sus datos y una vez contemos con este elemento, el siguiente paso será recolectar la información publicada en su perfil utilizando las herramientas o *scrapers* más adecuadas para cada ocasión. Con la información recopilada será necesario filtrar los datos y quedarse con lo que más nos interesa creando así la huella digital del usuario en nuestra investigación. Por último, para almacenar de forma correcta la información recolectada, se tendrá que analizar la estructura de la red social y crear modelos de datos que se adapten a las necesidades de cada una de ellas con el fin de mantener cada una de las investigaciones realizadas de manera persistente en una base de datos. Este procedimiento que se establece como una metodología de recopilación de la huella digital de un usuario, puede verse reflejada en el siguiente diagrama de secuencia UML.

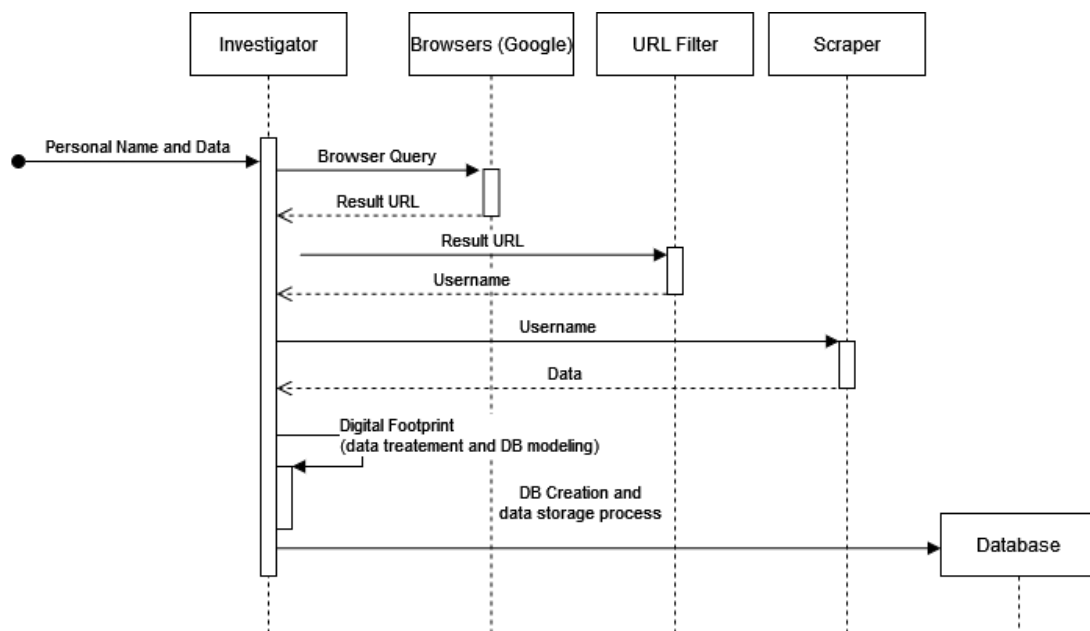


Figura 24. Diagrama de secuencia UML de la metodología de recolección de huella digital desarrollada. (Elaboración propia)

5.3 PREPARACIÓN DE LA DISTRIBUCIÓN

El último paso en el desarrollo de este proyecto consiste en la creación y configuración de la distribución que conformará el entorno de trabajo OSINT completo. A lo largo de este apartado se comentará el proceso de creación de esta distribución atendiendo a las principales decisiones tomadas. En los apéndices 9.2 y 9.3 se pueden consultar los manuales de instalación tanto de la aplicación web como de la distribución.

5.3.1 CREACIÓN Y CONFIGURACIÓN DE LA MÁQUINA VIRTUAL

Para la creación de la máquina virtual que se exportará como distribución se ha escogido el sistema operativo Debian 10 como base debido a que buscamos un entorno Linux estable que nos permite contar con las herramientas básicas de una máquina Linux para poder configurar el *framework* completo.

Una vez realizada la instalación del sistema operativo, crearemos un usuario genérico que permitirá a cualquiera que descargue la imagen acceder y modificar la máquina virtual según sus necesidades.

A continuación, instalaremos la funcionalidad Docker y descargaremos la aplicación web desarrollada para instalarla en el sistema de tal manera que el usuario pueda utilizarla nada más arrancar la distribución. La aplicación web también contará con un usuario creado específicamente para ser usado una vez se arranca la máquina de manera rápida y sencilla.

Por último, se instalarán las herramientas vistas en el apartado 2 de la memoria y se configurarán para dejarlas a punto para ser usadas. Estas herramientas se organizarán en directorios en función del enfoque de cada una de ellas y podrán ser accedidas desde el escritorio utilizando las barras de búsqueda de Debian.

Dado que algunas de las herramientas son sitios web que requieren ser accedidos con la ayuda de un navegador, se añadirán a marcadores los enlaces a estas páginas con el fin de agilizar su búsqueda.

Como conclusión, la distribución y la herramienta podrán ser descargadas en GitHub de forma independiente de tal manera que no sea necesario instalar la distribución para poder utilizar la aplicación web que se podrá ejecutar directamente en cualquier dispositivo con Docker instalado. El escritorio de la distribución creada puede verse en la figura 25.

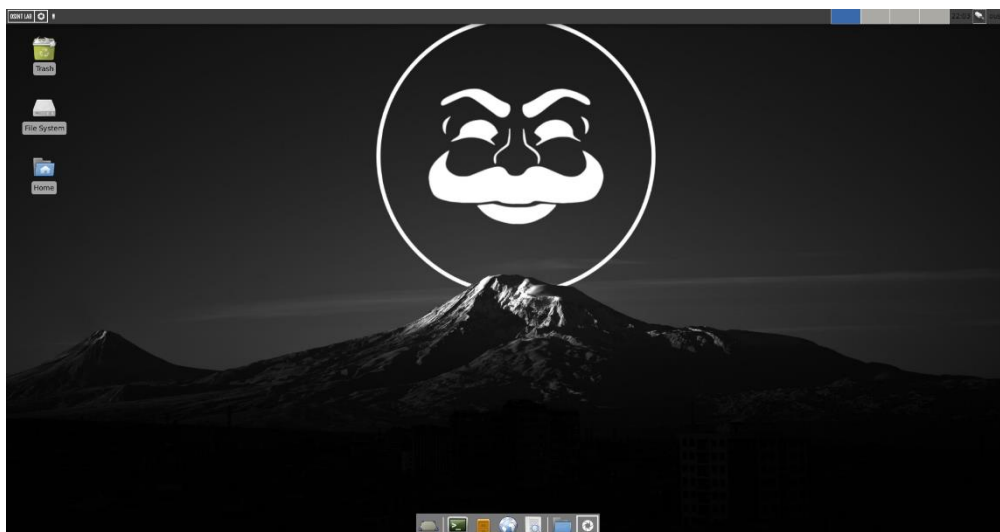


Figura 25. Escritorio de la distribución creada. (Elaboración propia).



6 RESULTADOS

Los resultados del trabajo conforman, además de la presente memoria, un conjunto de tres productos. En primer lugar, tenemos la metodología de búsqueda de *usernames* definida en el marco del presente trabajo que nos aporta un procedimiento a seguir en las investigaciones de fuentes abiertas.

Por otro lado, se ha implementado una herramienta web desarrollada con *frameworks* utilizados actualmente a nivel profesional como son Django Rest y React. La aplicación utiliza la metodología creada anteriormente para facilitar al usuario una interfaz sencilla de usar que recopila la huella digital del usuario a investigar en sus redes sociales. Los datos se almacenan utilizando una base de datos PostgreSQL siguiendo el modelado que puede verse representado en el diagrama entidad relación de la Figura 16.

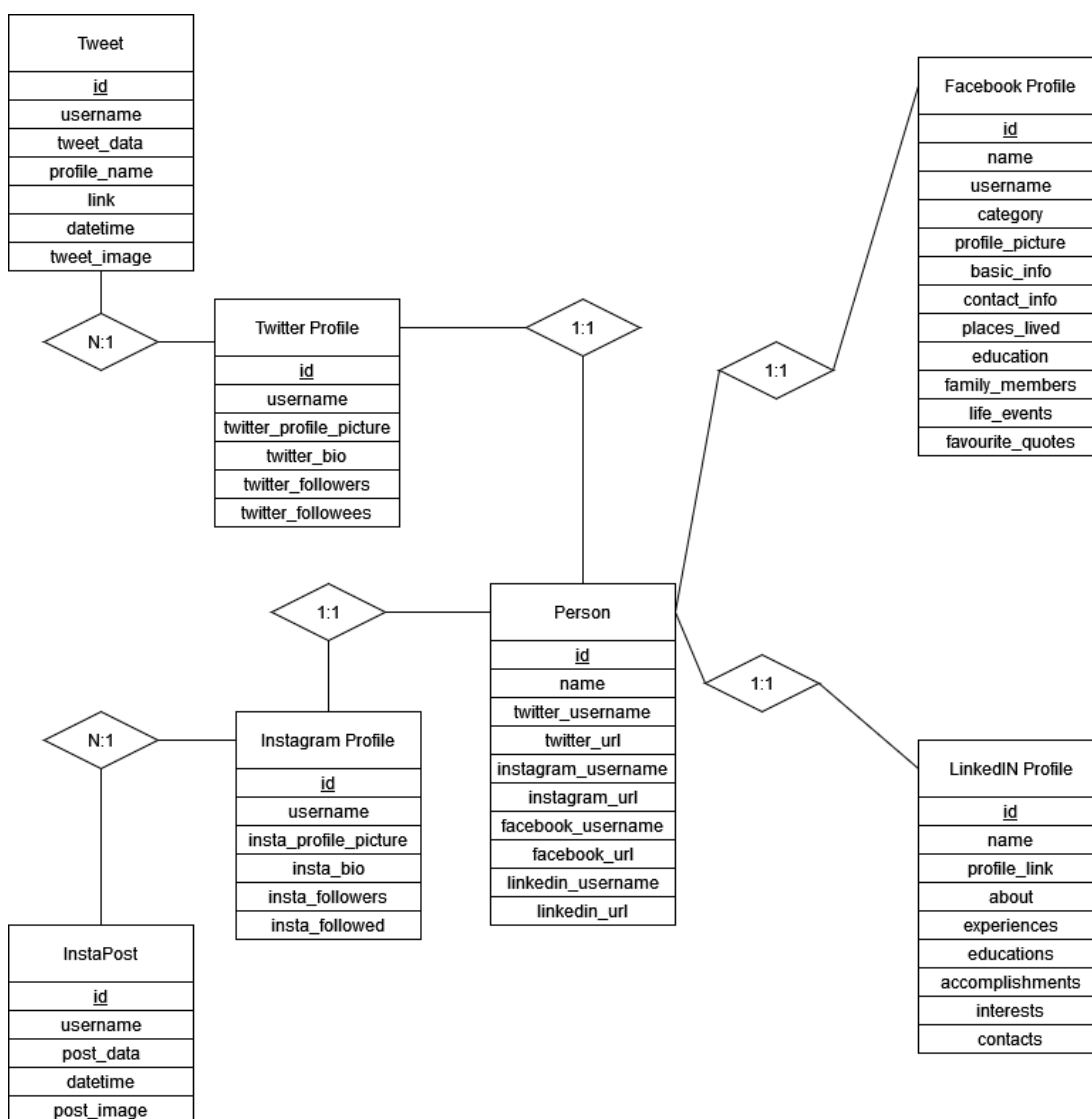


Figura 26. Diagrama entidad relación de la base de datos. (Elaboración propia)

La implementación de la lógica de la aplicación se ha dividido en distintos módulos o componentes que proporcionan interfaces entre sí para comunicarse y completar la operativa de la aplicación. En la Figura 17 puede verse el diagrama de componentes de la herramienta cuyo funcionamiento se ha descrito a lo largo del punto 5 del documento.

Además, la herramienta se despliega con el uso de contenedores Docker para otorgarle, independencia, portabilidad y facilidad de despliegue con respecto a la distribución desarrolla donde también se incluye la aplicación web.

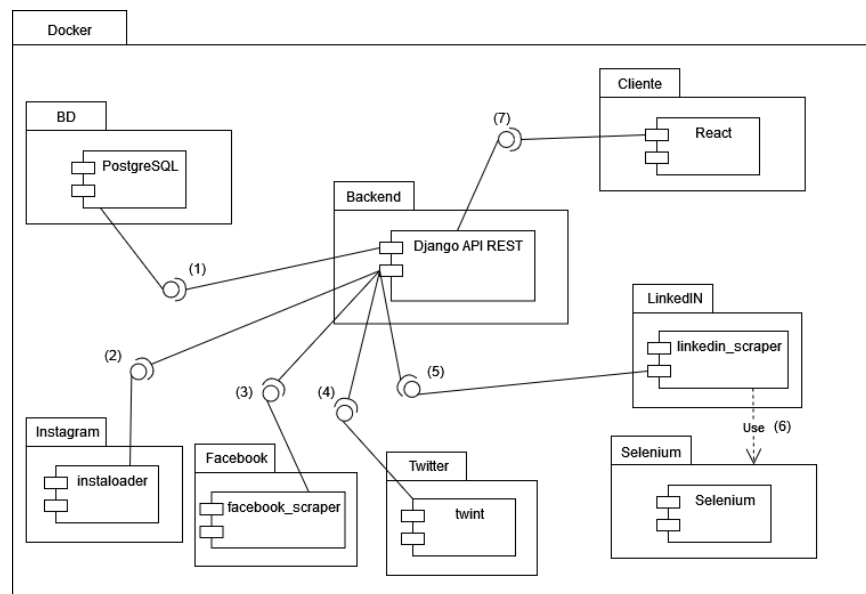


Figura 27. Diagrama de componentes de la herramienta. (Elaboración propia).

Por último, el producto final del desarrollo del proyecto se centra en la distribución de sistema operativo orientada al trabajo con tecnologías OSINT. Este entorno de trabajo cuenta con un set de herramientas que permiten agilizar el trabajo con las fuentes abiertas y la huella digital permitiendo a los investigadores contar con un *framework* centrado únicamente a OSINT, habiéndose desarrollado un conjunto de manuales para facilitar el uso reducir la curva de aprendizaje de los investigadores

Todo el trabajo resultante se publicará como proyecto *open source* en la plataforma GitHub para continuar con el desarrollo y mejora de las funcionalidades de la herramienta y permitir que pueda ser descargado por cualquier investigador que desee utilizar el entorno implementado. El código y los manuales pueden consultarse en el siguiente enlace: <https://github.com/jorgegene/OSINT-Framework.git>.

7 CONCLUSIONES

Con la realización del proyecto, los resultados nos han mostrado como el campo de las tecnologías y herramientas OSINT todavía tiene mucho camino por recorrer y por estandarizar, pero en la actualidad constituye una rama en continuo desarrollo que cada vez cobra más importancia en las empresas y organizaciones.

La primera parte del trabajo nos ha permitido descubrir que el origen de las técnicas OSINT reside en el interés de generar inteligencia para la toma de decisiones a partir de los datos públicos que podemos encontrar. En la actualidad, gran parte de los datos de una persona residen en sus redes sociales y tal y como se ha ido viendo, estos datos suponen una fuente abierta de información que pueden ser utilizados por empresas, organismos e investigadores para mejorar los servicios que prestan.

Las principales redes sociales utilizadas a día de hoy cuentan con estructuras basadas en perfiles y publicaciones que les permiten organizar a sus usuarios y presentar la información sobre el individuo. Esta información se ha podido recolectar utilizando *scrapers* que hacen uso de los sitios web desplegados para la visualización del contenido de un usuario extrayendo los datos que allí encuentra. Como hemos comprobado, en muchas ocasiones estas herramientas carecen en muchos casos de documentación y están desactualizadas lo que nos lleva a que, para realizar este tipo de investigaciones, sea necesario probar varias de ellas en cada una de las redes sociales y hasta encontrar la que mejor se adapta al desarrollo. También se ha comprobado que para la realización de una investigación en las redes sociales el elemento esencial es el nombre de usuario del individuo a investigar, elemento que en muchas ocasiones será complicado de obtener. La mejor forma de descubrir el nombre de usuario con el que se identifica una persona en cualquier red social vendrá dada por el uso de los buscadores más convencionales que nos permiten acotar los resultados hasta el punto de encontrar la persona buscada.

Por último, tras investigar las distintas alternativas a entornos de desarrollo OSINT, se ha podido comprobar que no son muchas las opciones existentes y menos en materia de redes sociales por lo que la estandarización y creación de nuevos *frameworks* que ayuden en el trabajo con fuentes abiertas será de gran ayuda en el desarrollo de nuevos proyectos sobre las tecnologías OSINT.

8 TRABAJOS FUTUROS

Son muchas las ideas que han ido surgiendo con el desarrollo del proyecto sobre futuros desarrollos o mejoras a implementar en la aplicación desarrollada. Algunos de los posibles trabajos futuros son los siguientes:

- a) Desarrollo de un sistema de geolocalización de las publicaciones de las distintas redes sociales para realizar un seguimiento del usuario. La mayoría de las redes sociales que se han investigado en la realización del proyecto, cuentan con un campo de geoposicionamiento en sus publicaciones. Con la recopilación de estos datos podría dibujarse sobre un mapa el recorrido realizado por una persona en relación a la información publicada en las redes.
- b) Uso de técnicas de inteligencia artificial para mejorar el procedimiento de búsqueda de *usernames* en función del nombre de la persona a investigar. Ya que este es uno de los temas más complejos, una de las posibilidades que se presentan es buscar alguna forma de afianzar los resultados de la búsqueda con la ayuda de campos como el Deeplearning o el análisis de imágenes. Esto nos permitiría relacionar nombres de usuario de distintas redes cuando existiera un alto grado de coincidencia entre la información recopilada.
- c) Búsquedas de mayor complejidad con el objetivo de facilitar la investigación OSINT. Son muchos los campos que se pueden filtrar a la hora de realizar búsquedas utilizando los *scrapers* que se han seleccionado, de esta manera se podría limitar el rango temporal de publicaciones a recopilar, búsqueda por palabras clave o por relación con otros usuarios.

Por otro lado, dado que el tema que nos concierne está en continuo cambio, y sobre todo en cuanto a las redes sociales por las modificaciones en sus sistemas, como trabajo futuro sería necesario mantener continuamente actualizado el entorno tanto a nivel de distribución descargando las actualizaciones pertinentes como a nivel de la aplicación web desarrollado para evitar que algunas de las funcionalidades dejen de funcionar.

9 APÉNDICES

9.1 BIBLIOGRAFÍA

BBC NEWS. (2009, 15 junio). *BBC NEWS / Middle East / Internet brings events in Iran to life*. BBC. http://news.bbc.co.uk/2/hi/middle_east/8099579.stm

Bean, H. (2011). *No More Secrets: Open Source Information and the Reshaping of U.S. Intelligence*. Praeger.

Christie, T. (2011). *Home - Django REST framework*. Django REST Framework. <https://www.django-rest-framework.org/>

Colquhoun, C. (2020, 17 julio). *A Brief History of Open Source Intelligence*.

Bellingcat. <https://www.bellingcat.com/resources/articles/2016/07/14/a-brief-history-of-open-source-intelligence/>

CoreUI. (2021). *GitHub - coreui/coreui-free-react-admin-template*. GitHub. <https://github.com/coreui/coreui-free-react-admin-template>

Datalux. (2021, 2 junio). *GitHub - Datalux/Osintgram*. GitHub. <https://github.com/Datalux/Osintgram>

Europol: Internet Organised Crime Threat Assessment (IOCTA) 2019. (2019). *Computer Fraud & Security, 2019*(11), 4. [https://doi.org/10.1016/s1361-3723\(19\)30114-9](https://doi.org/10.1016/s1361-3723(19)30114-9)

Everstine, B. (2017, 8 agosto). *Farewell, Bones: Air Force finishes latest round of B-1B bomber retirements*. Air Force Times. <https://www.airforcetimes.com/news/your-air-force/2015/06/04/carlisle-air-force-intel-uses-isis-moron-s-social-media-posts-to-target-airstrikes/>

Facebook Inc. (2021). *React – Una biblioteca de JavaScript para construir interfaces de usuario*. React. <https://es.reactjs.org/>

- Garner, B. (2021, 12 abril). *Build a REST API in 30 minutes with Django REST Framework*. Medium. <https://medium.com/swlh/build-your-first-rest-api-with-django-rest-framework-e394e39a482c>
- Googlesearch. (2018). *Welcome to googlesearch's documentation!* — googlesearch documentation. <https://python-googlesearch.readthedocs.io/en/latest/>
- Hassan, N. A., & Hijazi, R. (2018). *Open Source Intelligence Methods and Tools: A Practical Guide to Online Intelligence* (1st ed.). Apress.
- Instaloader. (2021). *Instaloader — Download Instagram Photos and Metadata*. <https://instaloader.github.io/>
- jnordine. (2021). *OSINT Framework*. OSINT Framework Web. <https://osintframework.com/>
- Joeyism. (2021). *GitHub - joeyism/linkedin_scraper*. GitHub. https://github.com/joeyism/linkedin_scraper
- Kevinzg. (2021). *GitHub - kevinzg/facebook-scraper*. GitHub. <https://github.com/kevinzg/facebook-scraper>
- laramies. (2021). *GitHub - laramies/theHarvester: E-mails, subdomains and names Harvester - OSINT*. GitHub. <https://github.com/laramies/theHarvester>
- Le-tools. (2021). *Le-tools.com - DumpItBlue+*. DumpItBlue+. <https://le-tools.com/DumpItBlueExtension.html>
- Maltego. (2021). *Maltego*. <https://www.maltego.com/>
- NATO. (2001). *NATO Open Source Intelligence Handbook*. NATO.
- Octoparse. (2020). *5 Cosas que Debes Saber Antes de Scrapear Facebook*. <https://www.octoparse.es/blog/5-cosas-que-saber-antes-de-scraping-de-facebook>

OSINTUX. (2018, 9 julio). *Osintux / Distribución Linux inteligencia en fuentes
abiertas OSINT*. <https://www.osintux.org/>

Schaurer, F., & Störger, J. (2013). The Evolution of Open Source Intelligence
(OSINT). *Journal of U.S. Intelligence Studies*, 19(3), 53–56.
[https://www.afio.com/publications/Schauer Storger Evo of OSINT WINT
ERSPRING2013.pdf](https://www.afio.com/publications/Schauer_Storger_Evo_of_OSINT_WINTER_SPRING2013.pdf)

Seisdedos, C., & Aguilera Díaz, V. (2020). *Open Source INTelligence (OSINT):
Investigar personas e Identidades en Internet* (1.^a ed.). 0xWORD.

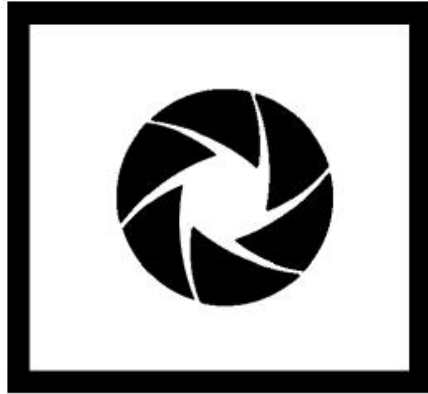
sherlock-project. (2021). *GitHub - sherlock-project/sherlock*. GitHub.
<https://github.com/sherlock-project/sherlock>

Shodan. (2021). *Shodan*. <https://www.shodan.io/>

Trace Labs OSINT VM. (2021). *OSINT VM*.
<https://www.tracelabs.org/initiatives/osint-vm>

Twintproject. (2021, 2 marzo). *GitHub - twintproject/twint*. GitHub.
<https://github.com/twintproject/twint>

Vivify-Ideas. (2021). *GitHub - Vivify-Ideas/python-django-drf-boilerplate*. GitHub.
<https://github.com/Vivify-Ideas/python-django-drf-boilerplate>



OSINT-Framework: Installation Guide

Version 1.0.0

Jorge Generelo
October 7, 2021



Contents

1	Introduction	2
2	OSINT-Framework OS	3
3	OSINT-Lab Web App	5
3.1	Facebook Cookie	5
3.2	LinkedIN Credentials	7
3.3	Docker Installation	7

1 Introduction

This document describes the steps to install the OSINT-Framework OS distribution and the OSINT-Lab web application.

The OS distribution requires Virtual Box to run and the following specifications on the host system:

- 4GB RAM
- 40 GB Free Storage
- 2 Core CPUs

The web app needs Git, Docker and docker-compose installed in the system. Already installed in the OS distribution.

2 OSINT-Framework OS

To install the OS on Virtual Box you need the ".ova" file that can be found in the GitHub repository. Once you have downloaded this VM image, go to Virtual Box and select the "Import Appliance" option and search the ova file downloaded (Figure 1)

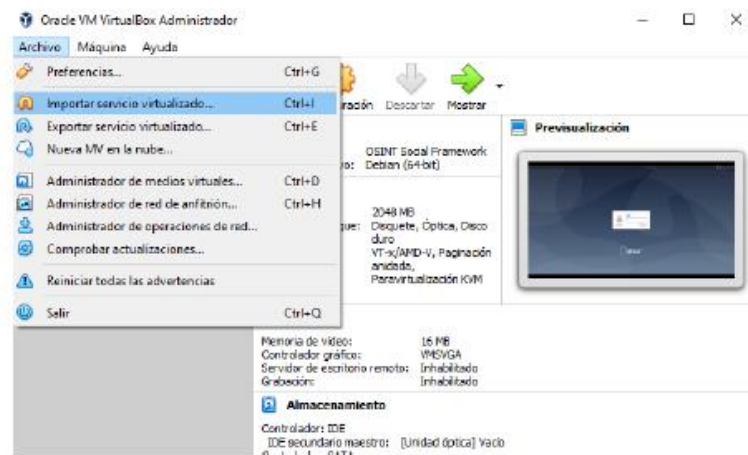


Figure 1: Import Virtual Box ova.

Run the new VM and once it starts, log in using the default credentials that can be found in the project repository (Figure 2).



Figure 2: Login screen.

After logging in the desktop will be displayed and the VM is ready to start a new investigation (Figure 3).



Figure 3: Desktop screen.

3 OSINT-Lab Web App

First step to install the Web App in our system is to download the repository using git and access the downloaded directory.

```
$ git pull https://github.com/jorgegene/OSINT-Framework.git  
$ cd OSINT-Framework
```

Once we are in the project folder there a few things we have to configure after installing the application.

3.1 Facebook Cookie

First of all we have to add a Facebook session cookie in order to allow the framework using our profile to do the searches (It is highly recommended not to use your personal profile). To get the cookie we can use the Firefox extension Cookie Quick Manager¹. Install the extension in your browser and log in to your Facebook account then select "Search cookies for facebook.com" (Figure 4).

¹<https://addons.mozilla.org/es/firefox/addon/cookie-quick-manager/>

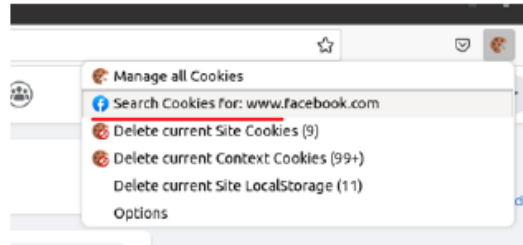


Figure 4: Search Facebook cookie.

Once we select the Facebook cookie we want to use download it using the export option and save it to a file named "cookie.json"(Figure 5).

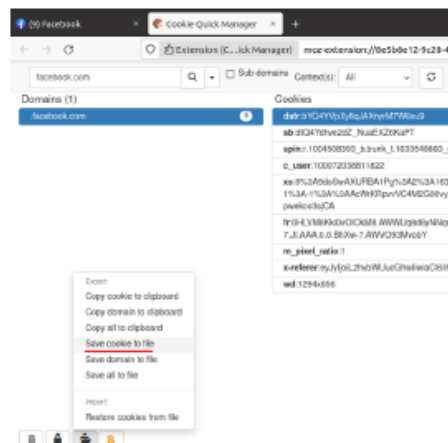


Figure 5: Export Facebook cookie.

Final step is to move the "cookie.json" file to the project root folder (an empty file with the same name can be found there).

3.2 LinkedIn Credentials

In order to allow LinkedIn search for profiles, we have to add the credentials of an existing account.

In this case just editing the ".env" file in the root folder adding your own credentials will make it work (Figure 6).

```
1 # TEMPORARY SECRET - DO NOT USE THIS
2 DJANGO_SETTINGS_MODULE=src.config.local
3 DJANGO_SECRET_KEY=
4 DJANGO_DEBUG=True
5
6 DB_NAME=database
7 DB_USER=user
8 DB_PASSWORD=password
9 DB_HOST=db
10 DB_PORT=5432
11
12 LINKEDIN_USER=
13 LINKEDIN_PASSWORD=
```

Figure 6: LinkedIn credentials.

3.3 Docker Installation

Once everything is ready run docker-compose up to deploy the containers and docker-compose up to run the containers. The build process just has to be done once.

```
$ sudo docker-compose build
$ sudo docker-compose up
```

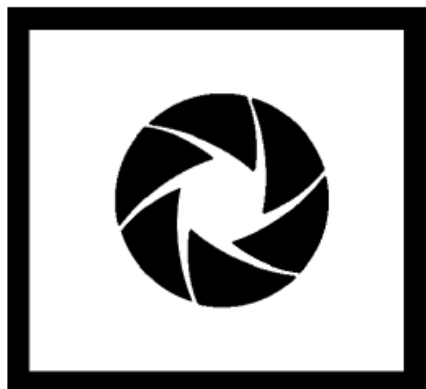
After deploying the environment access to the "api" docker container in order to create a new user on the framework using docker exec. The "api" container ID can be checked using docker ps.

```
$ sudo docker exec -it <container_ID> /bin/bash
```

In the container, use manage.py script to create a super user.

```
$ ./manage.py createsuperuser
```

Finally access to "http://127.0.0.1:9084/" using a web browser and log in the web application.



OSINT-Framework: User Guide

Version 1.0.0

Jorge Generelo

October 7, 2021

Contents

1	Introduction	2
2	OSINT-Framework OS	3
3	OSINT-Lab Web App	5

1 Introduction

This document describes the main features in the OSINT-Framework OS distribution and the OSINT-Lab web application.

If you need information about how to install the tools check the Installation Guide provided with the project.

2 OSINT-Framework OS

First of all, it is recommended to change the username password to a random one after the first login.

The provided OS image allows the user to work on an OSINT dedicated environment where you can find some of the most used tools. Once you have logged in the system the Desktop will appear showing icons that allow the user to get a shortcut to the OSINT tools installed. This icons are the two on the right of the bottom bar (Figure 1). The first one redirects to the folder where all the OSINT tools are installed and the second one to the path where de OSINT-Lab Web App is deployed.



Figure 1: OS Distribution Desktop.

The installed tools are classified to separate social network tools from pure OSINT frameworks using the folders as shown in Figure 2 (more tools would be added in the future).

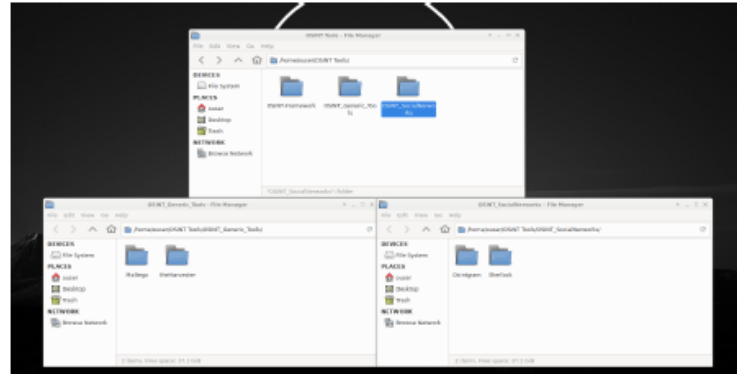


Figure 2: Installed OSINT tools.

Firefox, which is set as the default browser, will show you some links as bookmarks that could be helpful in your investigations (Figure 3).

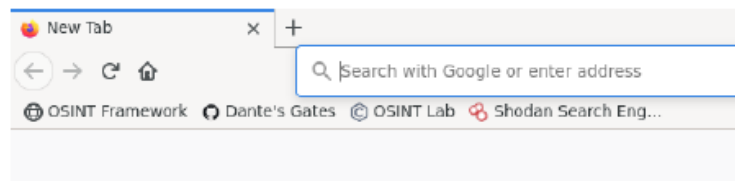


Figure 3: Firefox bookmarks.

3 OSINT-Lab Web App

The first step once the application has been deployed is to log in the web site using a created user¹ or registering a new one. Use a web browser to access "http://127.0.0.1:9084/" and the login screen will be displayed (Figure 4).

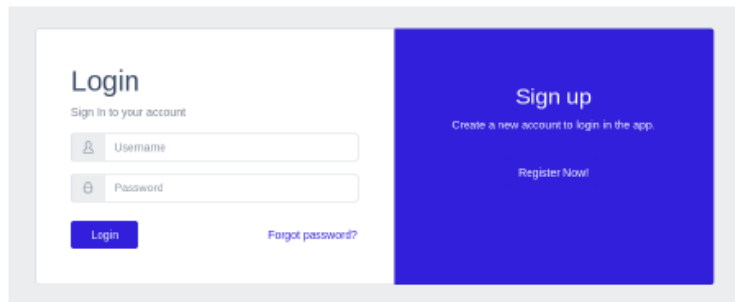


Figure 4: Login page.

Once you are logged in, the Home page will be prompted. In this place you can run a quick search using the personal name of the person you want to investigate (Figure 5). The quick search will check for social networks usernames of the person and recollect the digital footprint of the users found on Twitter, Instagram, Facebook and LinkedIn.

¹If you are working on the OSINT-Framework OS the credentials are provided in the project readme file.



Figure 5: Home page.

If you need an advanced search go to "ID Finder" tab and you will see a more complex search bar. In this page you can select which social networks you want to investigate and if you already know any username you can write them to avoid searching for them and getting better results (Figure 6).

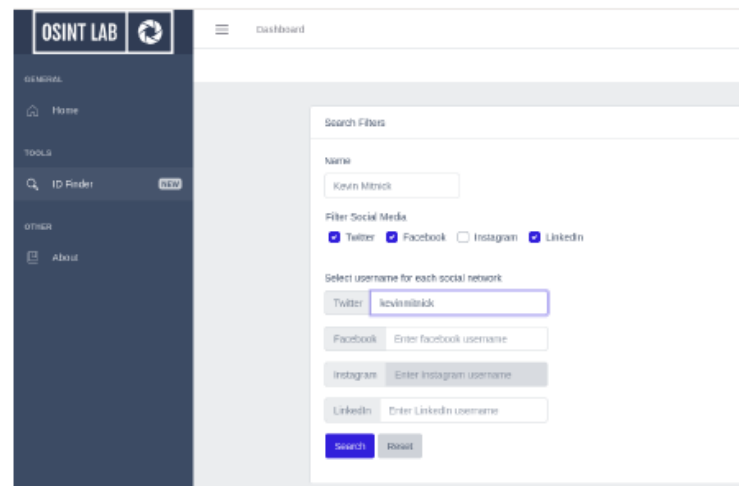


Figure 6: ID Finder page.

When the search is finished, you will be able to see the results of each one of the social networks in separated tabs. The framework will also prompt the usernames found linked to their websites in order to allow the investigator to visit the profile page where the data has been collected (Figure 7).

OSINT LAB

Dashboard

GENERAL

Home

TOOLS

ID Finder

OTHER

About

Search Filters

Name

Kevin Mitnick

Filter Social Media

☒ Twitter ☒ Facebook ☐ Instagram ☒ LinkedIn

Select username for each social network

Twitter kevinmitnick

Facebook Enter facebook username

Instagram Enter Instagram username

LinkedIn Enter LinkedIn username

Search Reset

Figure 7: ID Finder page.

9.4 GUÍA DE SOLUCIONADO DE ERRORES

Dado que las principales empresas que están detrás de las redes sociales buscan evitar que las prácticas como el *scraping* se realicen en sus sitios web, uno de los principales problemas que pueden surgir a la hora de utilizar las herramientas es el bloqueo de conexión por IP.

Para evitar que esto ocurra, durante el periodo de desarrollo se ha limitado el número de publicaciones recogidas de cada una de las redes sociales y se ha evitado realizar búsquedas de forma repetida en poco tiempo.

En el caso de que esto ocurra y no se recupere la información de un usuario buscado, la mejor solución hasta el momento consiste en el reinicio del *router* para obtener una nueva IP pública que no esté baneada en sus sistemas.

Es posible que si se realizan muchas peticiones en las webs que requieren el uso de una cuenta de usuario, el baneo se realice en función de la cuenta del usuario. En este caso no quedará más remedio que crear una nueva cuenta.

A última hora de la entrega del proyecto, la aplicación empezó a dar fallos en el tratamiento de los JWT Tokens. Tras investigar cual podía ser el error que ocasiono esto, ya que no se había realizado ningún cambio significativo, se descubrió que la versión 2.2.0 de PyJWT, lanzada el día 7 de Octubre de 2021 (un día antes de la fecha de entrega) estaba dando errores en un post de [stackoverflow](https://stackoverflow.com). Al modificar la versión a instalar en el fichero requirements.txt a la 2.1.0 la aplicación volvió a funcionar correctamente.

9.5 TECNOLOGÍAS UTILIZADAS

- **React:** biblioteca de JavaScript para construir interfaces de usuario.
- **Python:** lenguaje de programación interpretado.
- **Django Rest Framework:** conjunto de herramientas basado en Python y orientado a la creación de Web APIs.
- **GitHub:** repositorio de código fuente que permite el control de versiones mediante Git.
- **Docker:** proyecto de código abierto que automatiza el despliegue de aplicaciones dentro de contenedores de software.
- **Selenium:** Entorno de pruebas de software para aplicaciones basadas en la web. Permite simular una navegación web en navegador.
- **PostgreSQL:** también llamado Postgres, es un sistema de gestión de bases de datos relacional orientado a objetos y de código abierto.
- **Twint:** herramienta de *scraping* para la plataforma Twitter.
- **Osintgram e Instaloader:** herramientas de *scraping* para la plataforma Instagram.
- **Facebook_scraper y DumpItBlue:** herramientas de *scraping* para la plataforma Facebook.
- **Linkedin-scraper:** herramienta de *scraping* para la plataforma LinkedIn.
- **Axios:** cliente basado en HTTP para node.js.
- **Maltego:** software utilizado para la Inteligencia de fuentes abiertas y forense, desarrollado por Paterva.
- **Shodan:** motor de búsqueda que le permite al usuario encontrar iguales o diferentes tipos específicos de equipos conectados a Internet a través de una variedad de filtros.
- **theHarvester:** herramienta para la obtención de información en fuentes abiertas.
- **Sherlock:** herramienta OSINT que localiza perfiles de usuario a partir de un *username*.
- **Nginx:** servidor web/proxy inverso ligero de alto rendimiento.