



UNIVERSIDAD
POLITECNICA
DE VALENCIA



Máster Universitario
en Tecnologías, Sistemas y
Redes de Comunicaciones

Fuentes de Información OSINT para la Clasificación y Selección de Perfiles sobre Repositorios

Autor: Cristian Andrés Cuesta Martín

Director: Manuel Esteve Domingo

Fecha de comienzo: 02/03/2019

Lugar de trabajo: Grupo de Investigación Sistemas y Aplicaciones de Tiempo Real

Distribuidas

Departamento de Comunicaciones UPV

Objetivos — El presente trabajo tiene como objetivo principal adaptar una metodología OSINT, que consiste en aprovechar la enorme cantidad de información disponible que nos ofrece Github, con el fin de convertirla en inteligencia de datos, para mejorar el proceso de selección profesional de un grupo de desarrolladores registrados en la plataforma, mediante el uso de herramientas de software libre (XAMPP, HTML, CSS, JS, PHP,JSON, HTML).

Metodología — El desarrollo del proyecto se basó en el uso de herramientas de diseño web, plataforma de servidor y bases de datos. Como primera medida, se elaboraron los esquemas de la estructura de la página web con el lenguaje HTML, aplicando la parte visual con CSS, la parte interactiva con Javascript y operativa con un poco de PHP. Como segunda medida, para realizar el montaje de la página web, se hizo uso de Xampp y Apache, para hacer las pruebas pertinentes del código y que este fuera funcional. Como última medida, se hace uso de una base de datos MySQL para el almacenamiento y manejo de la información que se utiliza en los procesos de recopilación de los datos. Todo esto, organizado para dar forma a la arquitectura OSINT, permitiendo una administración de la información más eficiente.

Desarrollos teóricos realizados — Para esto, se analizó información pertinente a la metodología OSINT para poder entender cómo es el funcionamiento de este procedimiento. De manera que, por medio de los conocimientos adquiridos sobre sistemas, diseño web, tecnologías de redes y comunicaciones, las nuevas tendencias de inteligencia artificial en sistemas informáticos como machine learning y big data, permitiendo establecer una técnica para la realización del presente proyecto.

Desarrollo de prototipos y trabajo de laboratorio — Se creó la estructura del sitio web para la empresa, que permita detectar, analizar y organizar la información que se recopila de la big data pública, contenida en la internet, para realizar procesos de comparación, selección de perfiles, de acuerdo con los atributos y aspectos que se requieran evaluar en la acción pertinente. Esto se logró a lo largo de varias pruebas realizadas en el entorno de trabajo, codificación de lenguaje web, implementación de la plataforma del servidor y funcionamiento de la base de datos.

Resultados — En el desarrollo del proyecto, se fueron adquiriendo destrezas en la codificación de lenguaje web, debido a la experimentación y pruebas aplicadas, así como también, en el montaje de un servidor y de las bases de datos. Como producto final, la estructura del sitio web que es un paso para consolidar un servicio aplicado con respecto a la big data pública, luego de aplicar algunas mejoras más en el entorno gráfico y operativo.

Líneas futuras — Luego de los ensayos del proyecto, se prevee que la estructura de la página web se pueda aplicar a un servicio de comparación de perfiles para un buscador de empleo o en si mismo, un servicio a las empresas mismas en las ofertas laborales, determinando las características esenciales y eligiendo los participantes más aptos.

Resumen — Este proyecto fue desarrollado con el fin de prestar un servicio de análisis de big data para empresas. Se realizó la codificación de lenguaje web de la página en html, css, javascript, php. Se hizo el montaje del servidor por medio del programa xampp y del servicio apache. La base de datos que se manejó fue MariaDB.

Abstract — This project was developed in order to provide a big data analysis service for companies. The web language coding of the page was done in html, css, javascript, php. Server assembly was done through the xampp program and the apache service. The database that was managed was MariaDB.

Autor: Cristian Andrés Cuesta Martín, email: cricuem1@teleco.upv.es
Director: Manuel Esteve Domingo, email: mesteve@com.upv.es
Fecha de entrega: 10-11-19

INDICE

I.	INTRODUCCIÓN	4
II.	ESTADO DEL ARTE	5
II.1.	CIBER SEGURIDAD	5
II.1.1.	Conceptos de Ciberseguridad	5
II.2.	Ciberinteligencia.....	6
II.3	Proceso OSINT	9
III.	ARQUITECTURA PROPUESTA	11
IV.	HERRAMIENTAS Y RECURSOS INDISPENSANLES	12
IV.1	MySQL.	12
IV.2	Servidor Apache.	12
IV.3	Xampp.	13
IV.4	Html.....	13
IV.5	Css.	13
IV.6	Javascript.....	14
IV.7	Php.	14
IV.8	Json.	14
IV.9	Github.	15
V.	Desarrollo del proyecto.	15
V.1.	Documentación.....	15
V.2	Implementación del proyecto.....	22
VI.	Resultados Adquiridos	28
VII.	Conclusiones	30
VIII.	LINEAS FUTURAS	30
IX.	AGRADECIMIENTOS	31
X.	Referencias	32

I. INTRODUCCIÓN

El fenómeno de Internet sobresale especialmente a raíz de tratarse de una herramienta que facilita a las personas el acceso a la navegación web, que con el transcurso de los años es cada vez más rápida, logrando obtener cantidades infinitas de información de diferente índole, con un costo beneficio relativamente bajo, donde los operadores de telecomunicaciones, centros de investigación y fabricantes crean nuevas tecnologías, con mayor eficiencia y desempeño a nivel técnico, para el uso diario de los usuarios, consiguiendo así una brecha tecnológica cada vez más sesgada con la humanidad, convirtiéndose en una necesidad a la cual puede acceder gran parte de la población a nivel mundial, que es el acceso a los datos.

Con la ayuda de Internet, la mayor parte de la inteligencia predictiva se puede obtener de fuentes de información públicas y que no son clasificadas, es decir información que es sensible pero que no está restringida por la ley o regulada para clases particulares de personas, es por eso que la revolución de las tecnologías de la información están haciendo que las fuentes abiertas sean cada vez más accesibles, omnipresentes y valiosas para un análisis de inteligencia, que es menos costoso al ser un bien intangible. De tal manera que el mundo actual está en medio de una explosión de la información, en redes sociales, páginas web, aplicaciones de los Smartphone que día a día recolectan y aumentan la cantidad de información de un mayor número de personas, de diferentes países, estratos sociales, y edades que de alguna u otra forma la disponibilidad de esta enorme cantidad información y datos en Internet está llevando a lo que se conoce hoy como paradoja moderna. Es por eso que se ha creado OSINT (Open Source Inteligent) que es el arte y la ciencia de la creación de apoyo a la toma de decisiones basado filosóficamente en la evidencia y la ética, utilizando sólo fuentes abiertas, métodos legales y éticos, que desde el punto de vista de la ingeniería, se requiere el acceso a fuentes de información en las siguientes categorías con los siguientes porcentajes para que se realice de forma eficiente: digitales (20%), fuentes analógicas (no más del 30 %) y fuentes humanas (50 %), así se ha desarrollado una amplia variedad de técnicas y tecnologías para recopilar datos estructurados y no estructurados de fuentes abiertas, es por eso que durante más de 10 años Open Sources Intelligence (OSINT) ha estado empleando esta cantidad masiva de datos con el principal objetivo de permitir decisiones fundamentadas y acertadas con el análisis de los datos, más allá de la intuición y la experiencia humana.

A lo largo de la tesina se han identificado dos desafíos principales a abordar con el fin de extraer conocimiento de los datos obtenidos: El primero es, la búsqueda y el descubrimiento de información relevante en los datos no estructurados proporcionados por la Web. El segundo es, el análisis de la información recolectada, mostrándolo a través de una plataforma para tomar decisiones.

II. ESTADO DEL ARTE

II.1. CIBER SEGURIDAD

Los acontecimientos recientes acerca de la fuga de información, las noticias de atacantes informáticos doblegando protocolos y tecnologías de seguridad como en el caso de la compañía Telefónica y otras empresas top de diferentes sectores en al menos 99 países han sufrido un ciberataque (Sánchez, 2017), comprobando las fallas de seguridad que han presentado tanto en el sector público como en el sector privado, son argumentos suficientes para evidenciar que estamos en un nuevo escenario de riesgos y amenazas, donde la información se convierte en un arma estratégica y táctica, que cuestiona la gobernabilidad de una organización o la de una nación (Cano, 2011).

La ciberseguridad es el vínculo entre las herramientas a nivel político, seguridad, gestión de riesgos, acciones, formación, prácticas éticamente idóneas, de tal manera que los dispositivos y tecnologías que se encuentren disponibles puedan utilizarse con el objetivo de proteger un bien intangible como los activos de información dentro de la organización y los usuarios que interactúan en el ciberentorno (ITU, 2010).

Es decir, especifica los conceptos que gobiernan la seguridad a través del acceso a Internet, en cuanto a la información bancaria, claves, compras realizadas por usuarios a nivel mundial online, información médica, datos personales suministrados a entidades terceras, etc. Evidenciando la cantidad infinita de datos que diariamente circula por Internet es abrumadora, lo cual hace que la información como bien intangible pueda estar en las manos de las personas menos apropiadas, reflejando los riesgos a los cuales están expuestos la gran mayoría de la población, empresas, instituciones e incluso países. Sin dejar atrás que con el paso de los días el ciberentorno cada vez se hace más grande, de tal manera que la ciberseguridad debe garantizar el acceso fehaciente de los movimientos en la red.

II.1.1. Conceptos de Ciberseguridad

- *Vulnerabilidad:* Se dice que un sistema es vulnerable al contar con una especie de “agujero” para infiltrarse y causar daños. Este tipo de escenarios están provocados por errores de programación, diseño y estructura a nivel de arquitecturas, que permiten a un tercero realizar acciones maliciosas de diferente índole, causando daños de forma remota.
- *Disponibilidad:* Es la capacidad que debe tener un sistema para asegurar la continuidad de los servicios en un ciberentorno, así se genere un fallo, esto solo se puede lograr implementando las estrategias adecuadas, basadas en soluciones y arquitecturas que permitan soluciones de alta disponibilidad.

- *Integridad* : Implica mantener la exactitud, consistencia y fiabilidad de los datos, es decir la información no debe ser modificada, durante el ciclo de vida y tránsito entre distintos dispositivos, mediante medidas y algoritmos que logren asegurar que los datos no sean alterados por personas no autorizadas.
- *Confidencialidad*: Son las medidas adoptadas para garantizar la privacidad de la datos e impedir que la información sensible llegue a las personas equivocadas, así el acceso debe ser restringido, clasificar diferentes tipos de prioridades dentro del sistema de acuerdo a la categoría de los administradores.

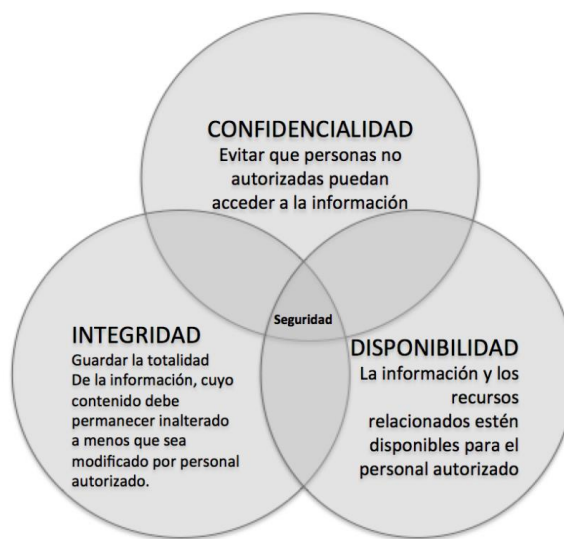


Figura 1. Ejes fundamentales de la ciberseguridad (Melo, 2014).

II.2. Ciberinteligencia

La inteligencia es el resultado de la recaudación, valoración, investigación, combinación e interpretación de la información disponible de fuentes internas como externas, que sean potencialmente significativas, en un determinado momento con el objetivo de permitir conocimiento de una situación específica, de tal manera que resulte útil al usuario en el momento de tomar decisiones independientemente del contexto, con un nivel menor de incertidumbre y con un mayor grado de efectividad, pues de esta dependen en cierta medida las fuentes de información.

La inteligencia tiene dos características críticas que lo distinguen de la información. La primera es que la inteligencia permite prever o predecir situaciones y circunstancias futuras, la segunda es que la inteligencia proporciona al usuario que se encuentre en mando una variedad de evaluaciones y estimaciones que facilitan la comprensión de la información. De tal manera que las evaluaciones son situacionales, por ejemplo, algunas evaluaciones estarán basadas en amenazas proporcionando

un análisis de las capacidades y las intenciones de las amenazas; Otros están basados en la población, proporcionando al usuario al mando un análisis de los factores socioculturales. Con estimaciones de inteligencia predictivas, precisas y pertinentes, los comandantes obtienen una ventaja de acuerdo con el contexto al tener la capacidad de comprender el ciclo de toma de decisiones de un adversario y, posiblemente, predecir y contrarrestar las operaciones adversarias. De tal manera que independientemente de la situación, las evaluaciones y estimaciones de inteligencia permiten a los comandantes formular planes y tomar mejores decisiones basándose en este conocimiento. Por lo tanto, la inteligencia predictiva, precisa y pertinente puede mitigar los riesgos inherentes a las operaciones de inteligencia y aumentar la probabilidad de éxito (Department Of The Army - United States Of America, 2013).

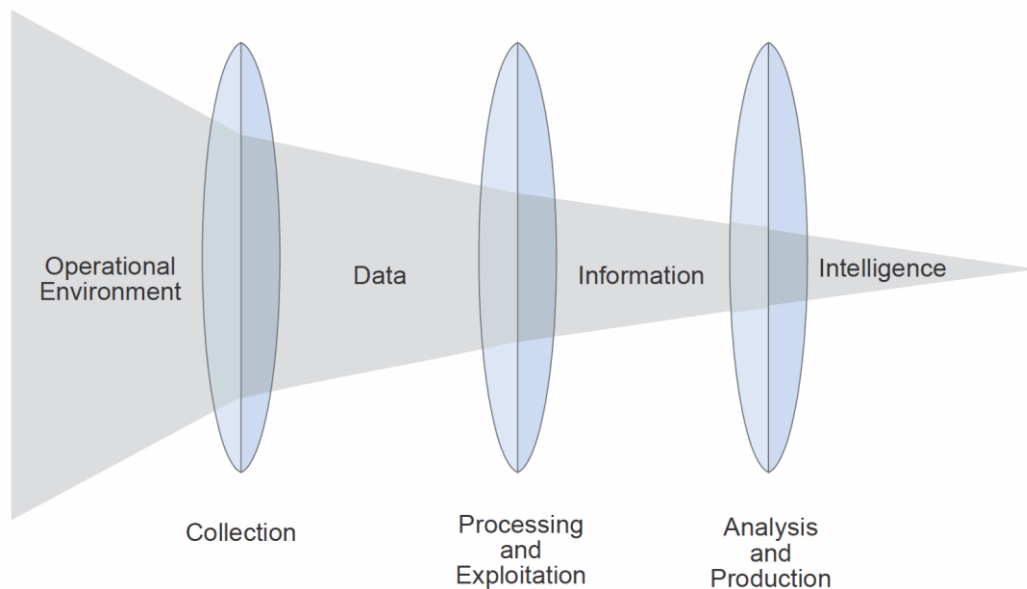


Figura 2. Relación de datos, información e inteligencia, (Department Of The Army - United States Of America, 2013)

II.2.1. Atributos de la Excelencia en Inteligencia

La calidad de la información de inteligencia es primordial para la capacidad del usuario al mando de para alcanzar y mantener la credibilidad de las acciones que se toman en determinados momentos a lo largo del tiempo, pues se deben reflejar en resultados positivos y tiempos medios de respuesta y esto depende en gran medida a los atributos de la calidad de inteligencia, de tal manera que para llegar a cierto nivel de excelencia deben contar con las siguientes características:

- **Anticipado:** La inteligencia debe anticipar las necesidades informativas del usuario al mando para guiar de forma proactiva a todos los recursos de forma conjunta, con el objetivo de proporcionar una base sólida de planificación operacional y toma de decisiones de la forma más efectiva.

- **Oportuno:** La fase de inteligencia debe estar disponible cuando el usuario al mando lo requiera, permitiendo anticiparse a los eventos de manera eficaz con el objetivo de evitar ser sorprendido.
- **Preciso:** La inteligencia debe ser correcta desde el punto de vista objetivo y subjetivo, es decir retransmitir la situación tal como realmente existe y proporcionar una comprensión de la situación basada en el juicio racional de la información disponible, este juicio debe tener en cuenta el engaño de una determinada situación, previamente de los datos obtenidos, evaluando la fiabilidad de la fuente mediante un proceso de retroalimentación en el que los datos recibidos se comparan con la “verdad del terreno”, por ejemplo cuando eventos o información posterior confirman la exactitud de la fuente, se continúa de forma asertiva por ese camino.
- **Usable:** La inteligencia debe adaptarse a las necesidades específicas del usuario al mando, para una comprensión inmediata de la situación para equilibrar la eficiencia y la efectividad.
- **Completa:** La inteligencia completa responde a la mayoría de las preguntas del adversario e informa al usuario al mando los aspectos relevantes de la situación.
- **Pertinente:** La inteligencia debe ser relevante para la planificación y ejecución de determinada situación, con el objetivo de ayudar al usuario al mando a entender el contexto.
- **Objetivo:** Debido al impacto decisivo y consecuente de la inteligencia sobre las operaciones y la dependencia de las decisiones de planificación y operaciones sobre la inteligencia, es importante mantener la objetividad y la independencia en el desarrollo de las evaluaciones.
- **Disponible:** La disponibilidad es una función no sólo de puntualidad y usabilidad, sino también de clasificación de seguridad, interoperabilidad y conectividad apropiadas.

II.3 Proceso OSINT

El proceso OSINT consta de las siguientes fases:

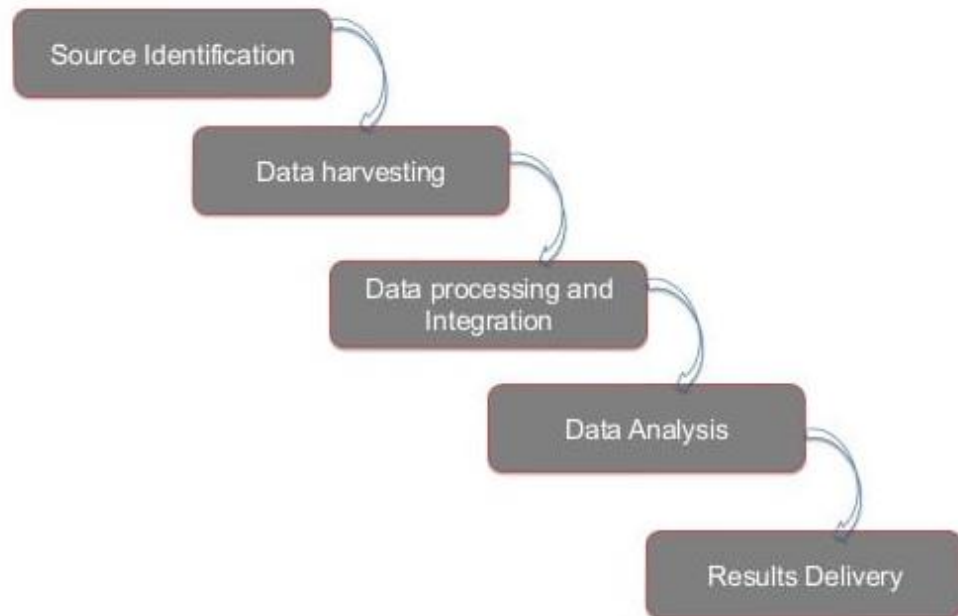


Figura 3. Esquema Proceso OSINT

- **Recolección de datos:** En esta fase, se recolecta información de las fuentes seleccionadas y se puede dar el escenario que se descubren fuentes nuevas a lo largo de esta fase.
- **Procesamiento e integración de datos:** durante esta fase, se procesa la información recolectada para procesarla mediante la búsqueda de información que pueda ayudar en la enumeración mediante filtros.
- **Análisis de datos:** Se realiza un análisis de datos de la información procesada que este arrojando la plataforma.
- **Entrega de resultados:** en la fase final, el análisis OSINT se completa y los resultados se presentan o se informan de la manera más adecuada

Todo el procedimiento puede ser visto como un proceso de obtención de información, de tal manera que el objetivo principal de la extracción de Información es transformar texto no estructurado en datos estructurados adecuados para almacenar en una base de datos.

Una herramienta de análisis OSINT también debe ser capaz de manejar la entrada manual de información en línea, informes e inteligencia derivados de fuentes abiertas o cerradas dependiendo del escenario en el cual se ponga a prueba, como los que se plantearan a continuación:

- **Redes sociales:** La explosión en el contenido en línea generado por las redes sociales y la mensajería en vivo con aplicaciones como Twitter, Facebook, Instagram y en cierta medida Youtube está cambiando la cara de la cobertura de los medios de comunicación, como los usuarios acceden a ella y cual satisface todas sus necesidades en determinado momento. Dicho contenido suele ser transitorio y pertinente sólo para la supervisión en tiempo real de los acontecimientos a medida que se desarrollan. Especialmente adecuados para este propósito son los motores de búsqueda en tiempo real que pueden proporcionar una vista de pájaro de una emergencia que se despliega. Dos útiles API de búsqueda para el monitoreo de la situación son Google Realtime y Kurrently.com . Un buen ejemplo de un monitor de situación en vivo, basado en tweets de todo el mundo, se puede encontrar en Trendmaps . El principal inconveniente con los datos derivados de Twitter o Facebook es su confiabilidad cuestionable y por esta razón no son realmente adecuados para la indexación a largo plazo y el archivo en repositorios OSINT (Best, 2011).
- **Confianza y verdad:** ¿Tal vez el área más difícil en el nuevo mundo de las publicaciones masivas en Internet es saber en quién y qué creer? Grupos opuestos y grupos de presión política saben muy bien la ganancia propagandística de convencer al mundo de alguna atrocidad o debilidad de un oponente que está diseñado para promover su causa ya sea por exageración o, a veces, por falsificación. Un analista de OSINT debe ser capaz de identificar las fuentes de confianza y las que son menos veraces. Esta es la razón por la fiabilidad de las fuentes es muy importante. Las fuentes deben ser clasificadas de acuerdo a la confiabilidad y etiquetadas por tipo: independientes, gobierno, rebeldes, políticos, etc, de tal manera que un informe sobre un solo evento debe dividirse en: Informe de primera mano, rumores, opinión, duplicado, así del mismo evento puede surgir un cuadro equilibrado. Esto también es cierto para los prejuicios cuando se obtiene información segmentada para el proceso de toma de decisiones, debido a que se puede tender a creer inconscientemente lo que la mente quiere pensar, siendo uno de los puntos subjetivos dentro de un proceso que requiere mucha precisión. Por lo tanto, los editores capacitados deben seguir las mismas reglas consistentes a todos los informes para evitar sesgos no intencionados (Best, 2011).

III. ARQUITECTURA PROPUESTA

Identificada la problemática que genera el gran número de información en la red, se propone una arquitectura para el proceso que lleva desde obtener la información para un respectivo análisis que ayude a la toma de decisiones de una forma más eficiente, de tal manera que la importancia de la extracción de información y el etiquetado semiautomático de los informes con los metadatos se hace cada vez más importante a medida que aumenta la inundación de información, así el objetivo es ordenar la información con el fin de filtrar los informes que deben ser leídos por una computadora en lugar de una persona.

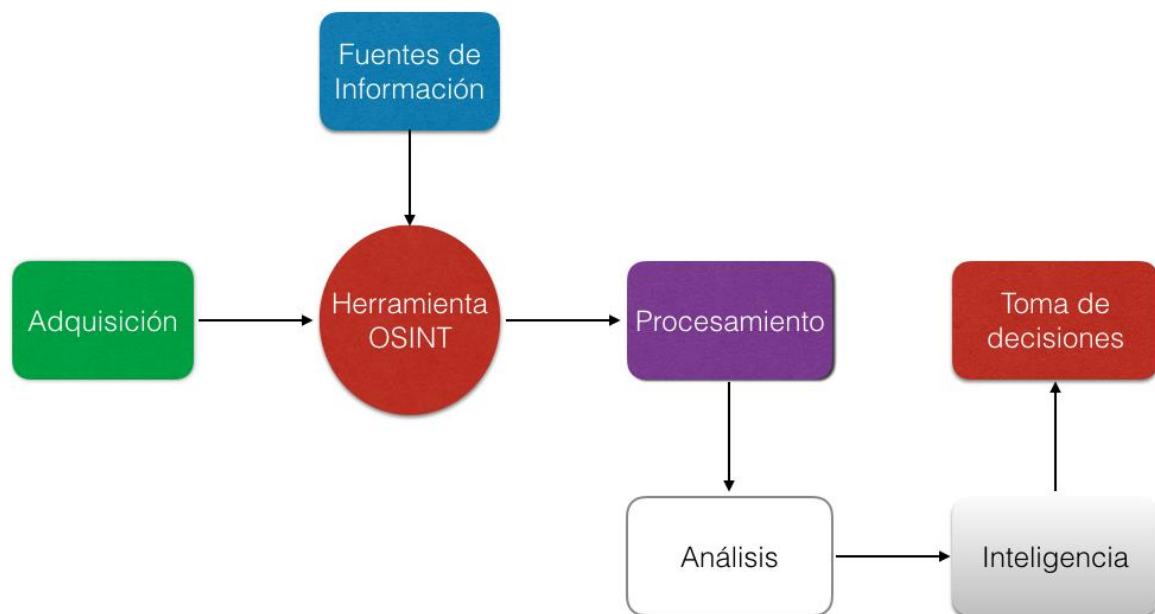


Figura 4. Arquitectura de bloques

- El primer desafío consiste en identificar y recuperar los datos más relevantes del mar de datos en Internet y en redes sociales como Github.
- El segundo desafío es clasificar y organizar esta información en orden de importancia según los datos obtenidos en Github de los desarrolladores, acorde a los filtros escogidos como (nombre, email, empresa, lugar de residencia, fecha de creación de cuenta, fecha de última conexión, repositorios públicos y número de seguidores), en donde se utilizaron técnicas especializadas de análisis de datos para identificar la información pertinente, recopilarla y luego filtrarla para su posterior procesamiento.
- Una vez recogida esta información en formato JSON, el tercer objetivo es almacenar y procesar la información en una base de datos, donde los registros son de los desarrolladores registrados en la red social Github.

- d) Una vez almacenada la información en la base de datos, se muestra a través de un dashboard en una tabla organizada por cantidad de repositorios de los desarrolladores registrados en github, cuales son los tres primeros que cumplen con los requisitos de selección, en donde el sistema de forma automática los selecciona y los marca para una lectura rápida de un color llamativo (verde).

IV. HERRAMIENTAS Y RECURSOS INDISPENSABLES

Para este desarrollo, se hizo uso de las siguientes herramientas de diseño web, como también de un repositorio en GitHub.

IV.1 MySQL.

Es un sistema de gestión de base de datos relacional (RDBMS) de código abierto, basado en lenguaje de consulta estructurado (SQL).

MySQL se ejecuta en prácticamente todas las plataformas, incluyendo Linux, UNIX y Windows. A pesar de que se puede utilizar en una amplia gama de aplicaciones, MySQL se asocia más con las aplicaciones basadas en la web y la publicación en línea y es un componente importante de una pila empresarial de código abierto llamado LAMP. LAMP es una plataforma de desarrollo web que utiliza Linux como sistema operativo, Apache como servidor web, MySQL como sistema de gestión de base de datos relacional y PHP como lenguaje de programación orientado a objetos (a veces, Perl o Python se utiliza en lugar de PHP).

MySQL, que fue concebido originalmente por la compañía sueca MySQL AB, fue adquirida por Oracle en 2008. Los desarrolladores todavía pueden usar MySQL bajo la Licencia Pública General de GNU (GPL), pero las empresas deben obtener una licencia comercial de Oracle. [1]

IV.2 Servidor Apache.

Es un software de servidor web gratuito y de código abierto con el cual se ejecutan el 46% de los sitios web de todo el mundo. El nombre oficial es Apache HTTP Server, y es mantenido y desarrollado por la Apache Software Foundation.

Permite a los propietarios de sitios web servir contenido en la web, de ahí el nombre de “servidor web”. Es uno de los servidores web más antiguos y confiables, con la primera versión lanzada hace más de 20 años, en 1995. [2]

IV.3 Xampp.

Es un paquete de instalación de software libre que consiste en un sistema de gestión de base de datos MySQL y de servidor Apache, así como de intérpretes de lenguajes PHP y Perl. De ahí que el nombre sea un acrónimo de X (refiriéndose a todos los lenguajes), Apache, MySQL, PHP y Perl.

En resumen, es una herramienta idónea para convertir nuestro pc en un servidor Apache y MySQL, convirtiendo nuestro equipo en un componente perfecto para el desarrollo de páginas web, o para la ejecución de cualquier software que requiera cualquiera de los componentes mencionados en las líneas anteriores. [3]

IV.4 Html.

Significa "Lenguaje de Marcado de Hipertexto" por sus siglas en ingles "HyperText Markup Language", es un lenguaje que pertenece a la familia de los "lenguajes de marcado" y es utilizado para la elaboración de páginas web. El estándar HTML lo define la W3C (World Wide Web Consortium) y actualmente HTML se encuentra en su versión HTML5.

Cabe destacar que HTML no es un lenguaje de programación ya que no cuenta con funciones aritméticas, variables o estructuras de control propias de los lenguajes de programación, por lo que HTML genera únicamente páginas web estáticas, sin embargo, HTML se puede usar en conjunto con diversos lenguajes de programación para la creación de páginas web dinámicas.

Básicamente el lenguaje HTML sirve para describir la estructura básica de una página y organizar la forma en que se mostrará su contenido, además de que HTML permite incluir enlaces (links) hacia otras páginas o documentos.

HTML es un lenguaje de marcado descriptivo que se escribe en forma de etiquetas para definir la estructura de una página web y su contenido como texto, imágenes, entre otros, de modo que HTML es el encargado de describir (hasta cierto punto) la apariencia que tendrá la página web. [4]

IV.5 Css.

Son las siglas de Cascading Style Sheets - Hojas de Estilo en Cascada - que es un lenguaje que describe la presentación de los documentos estructurados en hojas de estilo para diferentes métodos de interpretación, es decir, describe cómo se va a mostrar un documento en pantalla, por impresora, por voz (cuando la información es pronunciada a través de un dispositivo de lectura) o en dispositivos táctiles basados en Braille.

CSS es una especificación desarrollada por el W3C (World Wide Web Consortium) para permitir la separación de los contenidos de los documentos escritos en HTML, XML, XHTML, SVG, o XUL de la presentación del documento con las hojas de estilo, incluyendo elementos tales como los colores, fondos, márgenes, bordes, tipos de letra..., modificando así la apariencia de una

página web de una forma más sencilla, permitiendo a los desarrolladores controlar el estilo y formato de sus documentos.

El lenguaje CSS se basa en una serie de reglas que rigen el estilo de los elementos en los documentos estructurados, y que forman la sintaxis de las hojas de estilo. Cada regla consiste en un selector y una declaración, esta última va entre corchetes y consiste en una propiedad o atributo, y un valor separados por dos puntos. [5]

IV.6 Javascript.

Es un lenguaje que puede ser utilizado por profesionales y para quienes se inician en el desarrollo y diseño de sitios web. No requiere de compilación ya que el lenguaje funciona del lado del cliente, los navegadores son los encargados de interpretar estos códigos.

Este lenguaje posee varias características, entre ellas podemos mencionar que es un lenguaje basado en acciones que posee menos restricciones. Además, es un lenguaje que utiliza Windows y sistemas X-Windows, gran parte de la programación en este lenguaje está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas, cargas de páginas entre otros. [6]

IV.7 Php.

PHP (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.

En lugar de usar muchos comandos para mostrar HTML (como en C o en Perl), las páginas de PHP contienen HTML con código incrustado que hace "algo" (en este caso, mostrar "¡Hola, soy un script de PHP!"). El código de PHP está encerrado entre las etiquetas especiales de comienzo y final `<?php` y `?>` que permiten entrar y salir del "modo PHP".

Lo que distingue a PHP de algo del lado del cliente como Javascript es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente. El cliente recibirá el resultado de ejecutar el script, aunque no se sabrá el código subyacente que era. El servidor web puede ser configurado incluso para que procese todos los ficheros HTML con PHP. [7]

IV.8 Json.

Es un formato abierto de representación e intercambio de objetos, popularizado a principios de siglo por Douglas Crockford como una alternativa ligera a XML.

En lugar de usar muchos comandos para mostrar HTML (como en C o en Perl), las páginas de Una de sus principales ventajas es que permite intercambiar datos estructurados entre diferentes programas y medios de almacenamiento independientemente del lenguaje de programación en que

estén desarrollados. Además de todo, es fácil de generar y leer tanto por máquinas como por humanos. [8]

IV.9 Github.

Es un servicio de alojamiento que ofrece a los desarrolladores repositorios de software usando el sistema de control de versiones, Git.

Los repositorios son públicos, sin embargo, Github tiene una forma de pago que te permite alojar tus proyectos de forma privada. [9]

V. Desarrollo del proyecto.

Se hace detalle de cada uno de los procesos que componen la elaboración del presente proyecto, para dan una mejor comprensión de esta implementación.

V.1. Documentación.

A continuación, se muestra la organización de archivos y grupos de metadatos necesarios para la implementación.

En la Fig. 5, se muestra la estructura del sitio web que permite detectar, analizar y organizar la información de archivos Json de perfiles de Github para escoger al más calificado de acuerdo con la cantidad de repositorios.

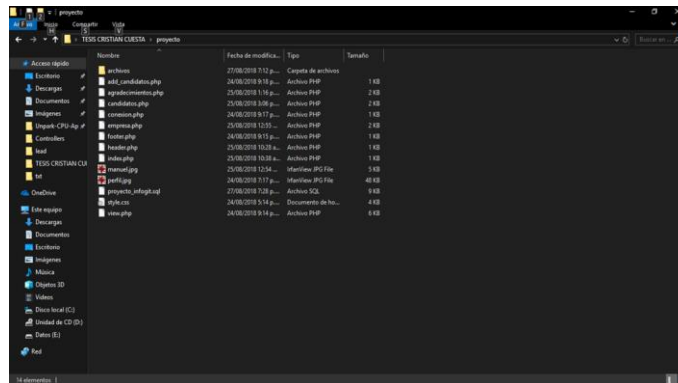


Figura 5. Estructura del sitio web.

En la Fig. 6, muestra el contenido de la carpeta *archivos* donde están alojados todos los archivos Json de los perfiles de Github que podrán ser visualizados y agregados a nuestra base de datos para ser organizados de acuerdo con la cantidad de repositorios.

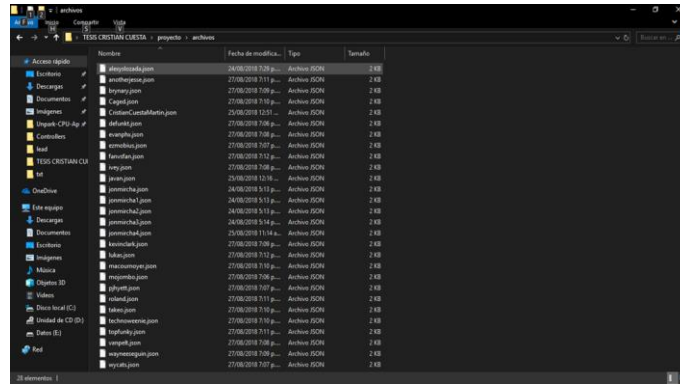


Figura 6. Carpeta archivos.

En la Fig. 7, muestra el archivo *add_candidatos.php*, el cual permite agregar a la base de datos la información del archivo Json que estemos analizando

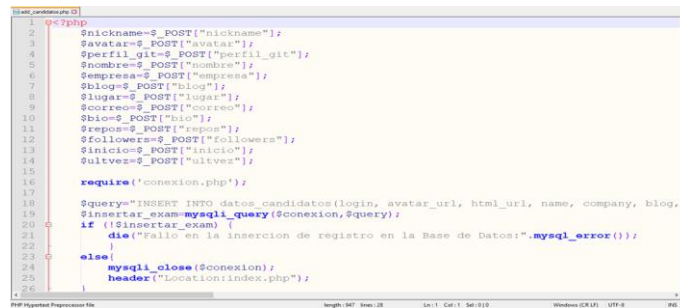


Figura 7. Archivo *add_candidatos.php*.

En la Fig. 8, muestra el archivo *agradecimientos.php*, donde está el código de la página de agradecimientos donde muestra la información del Ingeniero Manuel Esteve Domingo.



Figura 8. Archivo *agradecimientos.php*.

En la Fig. 9, muestra el archivo *candidatos.php*, que contiene el código donde se hace la consulta a la base de datos y los trae organizados de mayor a menor de acuerdo con el número de repositorios, una vez cargados se cierra la conexión.

```

1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4 <meta charset="UTF-8">
5 <meta http-equiv="Expires" content="0">
6 <title>Candidatos</title>
7 <link rel="stylesheet" href="style.css">
8 </head>
9 <body>
10 <?php
11 include("header.php");
12 //se hace una conexión con la base
13 require("conexion.php");
14 $a="";
15 //se consulta la tabla donde estan los datos y se ordena de mayor a menor numero de repociones
16 $query=$conexion->query("SELECT * FROM datos_candidatos ORDER BY public_repos DESC");
17 while($row_comp=$query->fetch_row()){
18 //se cargan las variables con los datos requeridos
19 $nombre[$a]=$row_comp[4];
20 $empresa[$a]=$row_comp[5];
21 $lugar[$a]=$row_comp[1];
22 $email[$a]=$row_comp[3];
23 $rep[$a]=$row_comp[10];
24 $segi[$a]=$row_comp[11];
25 $create[$a]=$row_comp[12];
26 $update[$a]=$row_comp[13];
27 $a++;
28 }
29 $query->free();
30 //se cierra la conexión
31 mysqli_close($conexion);
32 ?>
33 <div class="contenedor">

```

Figura 9. Archivo *candidatos.php*.

En la Fig. 10, muestra la parte del código donde los tres (3) primeros puestos se marcan en verde.

```

33 <div class="contenedor">
34 <table class="tbl-candidatos">
35 <caption class="caption">Estos son los candidatos organizados por mayor cantidad de proyectos en github
36 <thead>
37 <tr>
38 <th>Nombre</th>
39 <th>Email</th>
40 <th>Empresa</th>
41 <th>Lugar de residencia</th>
42 <th>Fecha creación de cuenta</th>
43 <th>Fecha última conexión</th>
44 <th>Repositorios Públicos</th>
45 <th>Seguidores</th>
46 <tbody>
47 <?php for($i=0;$i<$a;$i++) {
48 //se hace un filtro para los 3 que tengan mayor numero de proyectos, se colocan verde
49 if (($i==0) || ($i==1) || ($i==2)) [$aprobado="aprobados"]; else [$aprobado=""];
50 ?>
51 <tr class="?php echo $aprobado ??>
52 <td>?php echo $nombre[$i]; ?></td>
53 <td>?php echo $email[$i]; ?></td>
54 <td>?php echo $empresa[$i]; ?></td>
55 <td>?php echo $lugar[$i]; ?></td>
56 <td>?php echo $create[$i]; ?></td>
57 <td>?php echo $update[$i]; ?></td>
58 <td>?php echo $segi[$i]; ?></td>
59 </tr>
60 </tbody>
61 </table>
62 </div>
63 <?php include("footer.php"); ?>
64 </body>
65 </html>

```

Figura 10. Selección candidato *candidatos.php*.

En la Fig. 11, se muestra el código que maneja la conexión a la base de datos.

```

1 <?php
2 $hostname_conexion = "localhost";
3 $database_conexion = "proyecto_infogit";
4 $username_conexion = "root";
5 $password_conexion = "";
6
7 $conexion = new mysqli($hostname_conexion, $username_conexion, $password_conexion, $database_conexion);
8 if ($conexion->connect_errno) {
9 die("Fallo la conexión a MySQL: (" . $conexion->mysqli_connect_errno()
10 . ") " . $conexion->mysqli_connect_error());
11 }
12
13 ?>

```

Figura 11. Archivo *conexion.php*.

En la Fig. 12, muestra el código donde se encuentran los datos de la empresa que quisiera hacer uso de este sitio web, en este caso contiene mis datos personales.

```

1 <DOCTYPE html>
2 <html lang="es">
3 <head>
4 <meta charset="UTF-8">
5 <meta http-equiv="X-UA-Compatible" content="IE=edge">
6 <title>Repositorio</title>
7 <link rel="stylesheet" href="style.css">
8 </head>
9 <body>
10 <?php include("header.php"); ?>
11 <div class="contenedor">
12 <div class="ficha-suscriptor ajustes-usa-linea">
13 <div class="ficha-img ajustes-usa-linea">
14 <img alt="perfil.jpg" class="cristian-cuesta" class="img-avatar">
15 <div class="ajustes-usa-linea">
16 <span class="titu-em">NickName:</span>cristian.cuastemartin/>
17 </div>
18 <div class="ficha-datos ajustes-usa-linea">
19 <div class="ficha-f">
20 <div class="ficha-datos de Usuario">
21 <div class="ficha-f">
22 <div class="ficha-f">
23 <div class="ficha-f">
24 <div class="ficha-f">
25 <div class="ficha-f">
26 <div class="ficha-f">
27 <div class="ficha-f">
28 <div class="ficha-f">
29 <div class="ficha-f">
30 <div class="ficha-f">
31 </div>
32 </div>
33 </div>
34 <?php include("Footer.php"); ?>
35 </body>
36 </html>

```

Figura 12. Archivo empresa.php.

Código del footer con generador automático del año en curso en lenguaje php.

```

1 <?php echo date("Y"); ?>
2 <div class="copy">Todos los derechos reservados 2018 - <?php echo date("Y"); ?></div>
3 </div>

```

Figura 13. Archivo footer.php.

Código del header donde está el menú a las 4 páginas que contiene el sitio web.

```

1 <?php echo date("Y"); ?>
2 <div class="copy">Todos los derechos reservados 2018 - <?php echo date("Y"); ?></div>
3 </div>
4 <div class="copy">Todos los derechos reservados 2018 - <?php echo date("Y"); ?></div>
5 </div>
6 <div class="copy">Todos los derechos reservados 2018 - <?php echo date("Y"); ?></div>
7 </div>
8 <div class="copy">Todos los derechos reservados 2018 - <?php echo date("Y"); ?></div>
9 </div>
10 </div>

```

Figura 14. Archivo header.php.

Código de la página principal donde se listan los Json alojados en la carpeta de archivos ordenados por su nombre con la opción de un botón de view para poder escoger la opción de guardar la información en la base de datos.

```

1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4 <meta charset="UTF-8">
5 <meta http-equiv="Expires" content="0">
6 <title>Inicio</title>
7 <link rel="stylesheet" href="style.css">
8 </head>
9 <body>
10 <?php include("header.php"); ?>
11
12 <div class="contenedores">
13 <h1>Listado de archivos contenidos en la carpeta:</h1>
14 <?php
15 $path="archivos/";
16 $directorio=dire($path);
17 while ($archivo = $directorio->read()) {
18     if ($archivo!=".") {
19         $separar = explode(".", $archivo);
20         <form action="view.php" method="POST">
21             <input type="hidden" name="archivo" value="<?php echo $separar[0]; ?>">
22             <span class="lista-archivos ajustar-una-linea"><?php echo $separar[0]; ?></span>
23             <button class="view" onclick="this.form.submit()">Ver</button>
24         </form>
25     }
26 }
27 }
28 $directorio->close();
29 </div>
30 <?php include("footer.php"); ?>
31 </body>
32 </html>

```

Figura 15. Archivo index.php.

Foto usada para la página de agradecimientos del Ingeniero Manuel Esteve Domingo.

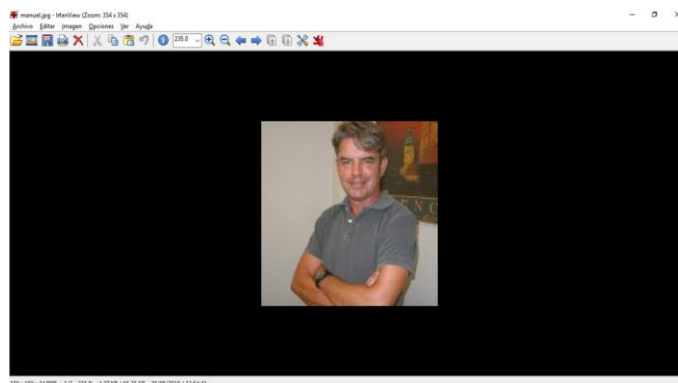


Figura 16. Archivo manuel.jpg.

Foto personal usada en la página de empresa.

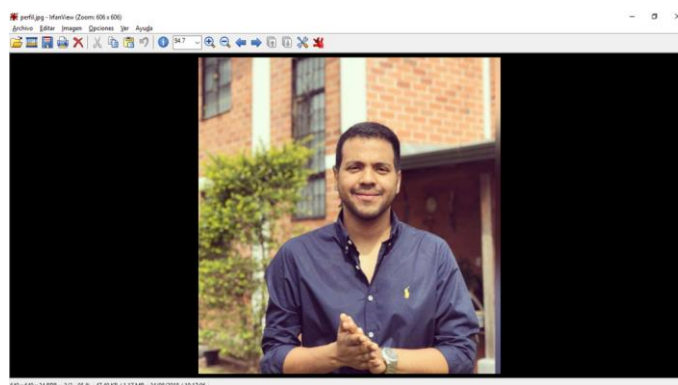


Figura 17. Archivo perfil.jpg.

Este archivo se usa para visualizar el archivo Json y decidir si se agrega esa información en la base de datos para ser un posible candidato o por el contrario darle clic al botón de volver escoger otros perfiles.

```

1 <!DOCTYPE html>
2 <html lang="es">
3 <head>
4 <meta charset="UTF-8">
5 <meta http-equiv="Expires" content="0">
6 <title>Candidatos</title>
7 <link rel="stylesheet" href="style.css">
8 </head>
9 <body>
10 <?php include("header.php"); ?>
11 <div class="contenedor">
12 <?php
13 $a="";
14 //se lee la variable
15 $archivo=$_POST["archivo"].".json";
16 //se abre el archivo que se escogio
17 $pagina=fopen($archivo,"r");
18 while (!feof($pagina))
19 {
20 $linea=fgets($pagina);
21 $linea=str_replace("\n","");
22 if($linea!="")
23 //se quitan algunos caracteres que no se necesitan
24 $linea=str_replace(' ','',$linea);
25 //se separan los textos para poderlos leer
26 $a=explode(":",$linea);
27 $item[]=$a;
28 if(!empty($a[1])||$a[1]=$a[2]) else($a[1]="");
29 if(!empty($a[2])||$a[2]=$a[3]) else($a[2]="");
30 $a="";
31 }
32 }

```

Figura 21. Archivo view.php, elegir perfil.

Después de abrir el Json de la carpeta de archivos y quitar los caracteres que no necesitamos separamos los textos para una mejor lectura, guardamos la información en un vector y lo recorremos escogiendo únicamente las variables que queremos guardar.

```

33 //se cierra la lectura del archivo
34 fclose($pagina);
35 //se recorre el vector
36 for($i=0;$i<count($item);$i++)
37 {
38 //se guardan las variables que se quieren manejar
39 if($item[$i]=="login"){$nickname=$a[1];} if($item[$i]=="avatar_url"){$avatar=$a[1];} if($i%2=="0"){$i%2=="1"}
40 if($item[$i]=="html_url"){$perfil_git=$a[1];} if($i%2=="0"){$i%2=="1"} if($item[$i]=="name"){$nombre=$a[1];}
41 if($item[$i]=="company"){$empresa=$a[1];} if($item[$i]=="blog"){$blog=$a[1];} if($i%2=="0"){$i%2=="1"}
42 if($item[$i]=="location"){$lugar=$a[1];} if($item[$i]=="email"){$correo=$a[1];}
43 if($item[$i]=="bio"){$biografia=$a[1];} if($item[$i]=="public_repos"){$repos=$a[1];}
44 if($item[$i]=="followers"){$followers=$a[1];} if($item[$i]=="created_at"){$inicio=$a[1];}
45 if($item[$i]=="updated_at"){$ultima=$a[1];}
46 }
47 }
48 //se le da formato a las variables por si estan vacias
49 if(empty($nickname)||$nickname==""){$avatar=""}; if(empty($perfil_git)||$perfil_git=="");
50 if(empty($nombre)||$nombre==""){$empresa=""}; if(empty($blog)||$blog=="");
51 if(empty($lugar)||$lugar==""){$correo=""}; if(empty($biografia)||$biografia=="");
52 if(empty($repos)||$repos==""){$followers=""}; if(empty($inicio)||$inicio=="");
53 if(empty($ultima)||$ultima=="");
54 }
55 <div class="ficha-usuario ajustar-una-linea">
56 <form action="" method="POST" name="form">
57 <input type="hidden" name="nickname" value="<?php echo $nickname; ?>">
58 <input type="hidden" name="avatar" value="<?php echo $avatar; ?>">
59 <input type="hidden" name="perfil_git" value="<?php echo $perfil_git; ?>">
60 <input type="hidden" name="nombre" value="<?php echo $nombre; ?>">
61 <input type="hidden" name="empresa" value="<?php echo $empresa; ?>">
62 <input type="hidden" name="blog" value="<?php echo $blog; ?>">
63 <input type="hidden" name="lugar" value="<?php echo $lugar; ?>">
64 <input type="hidden" name="correo" value="<?php echo $correo; ?>">
65 <input type="hidden" name="bio" value="<?php echo $biografia; ?>">
66 <input type="hidden" name="repos" value="<?php echo $repos; ?>">
67 <input type="hidden" name="followers" value="<?php echo $followers; ?>">

```

Figura 22. Archivo view.php, guardar información.

En esta parte cargamos la imagen y el nickname, luego cargamos los datos para mostrar y los dejamos en un formulario esperando si la información será guardada o descartada dependiendo del botón que se oprima.

Se inician los servicios de Apache y MySQL.

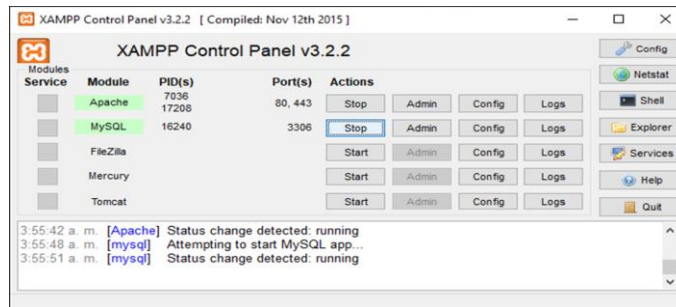


Figura 26. Inicio de servicios Apache y MySQL.

Se guarda la carpeta proyecto en el directorio htdocs del directorio Xampp.

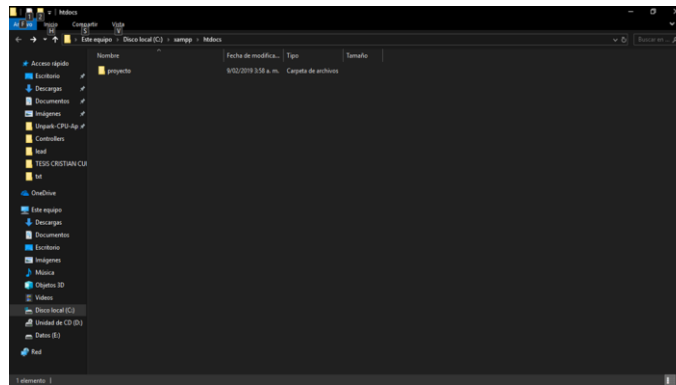


Figura 27. Carpeta htdocs del directorio Xampp.

Se da click en el botón de Admin del servicio MySQL.

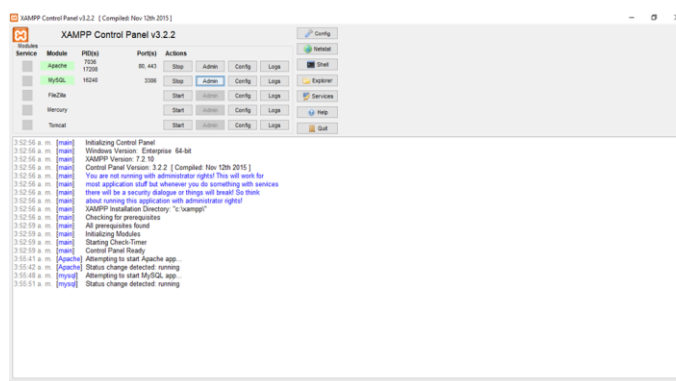


Figura 28. Botón Admin servicio MySQL.

Se abre el phpMyAdmin que es el gestor de bases de datos del XAMPP.

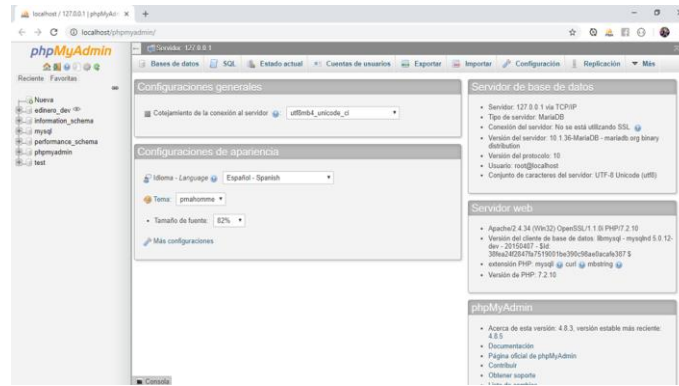


Figura 29. Gestor de base de datos phpMyAdmin.

Se crea la base de datos con el nombre proyecto_infogit con el cotejamiento utf8_unicode para no tener problemas con tildes y caracteres del idioma español.

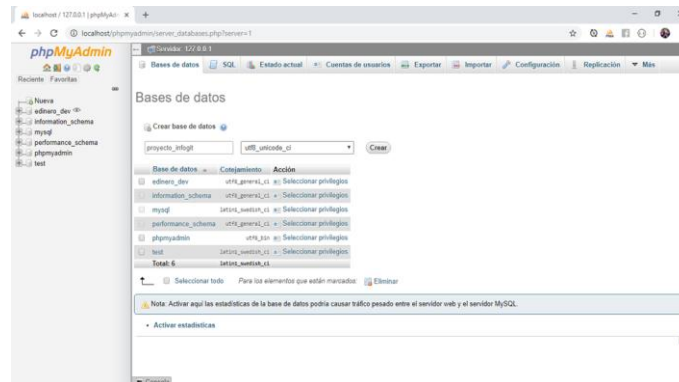


Figura 30. Crear base de datos.

Una vez creada la base de datos, se da click en la pestaña seleccionar archivo para importar la base de datos.

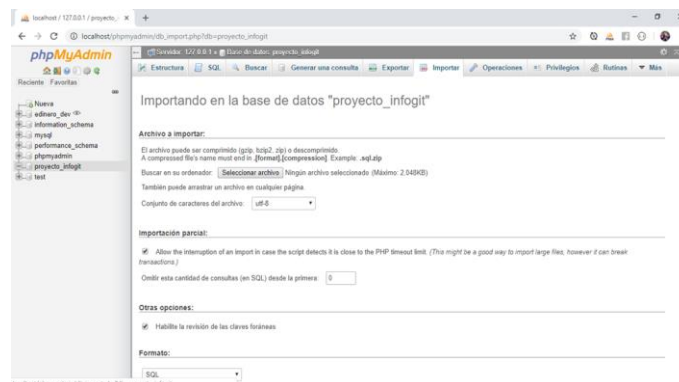


Figura 31. Importar archivo base de datos.

Se busca dentro de la carpeta del proyecto el archivo proyecto_infogit.sql, el cual se selecciona y se procede a darle abrir.

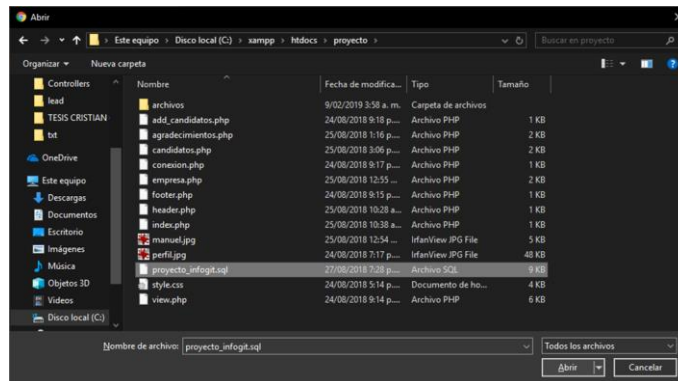


Figura 32. Selección del archivo proyecto_infogit.sql.

Después de seleccionar el archivo correcto, se da click en continuar.

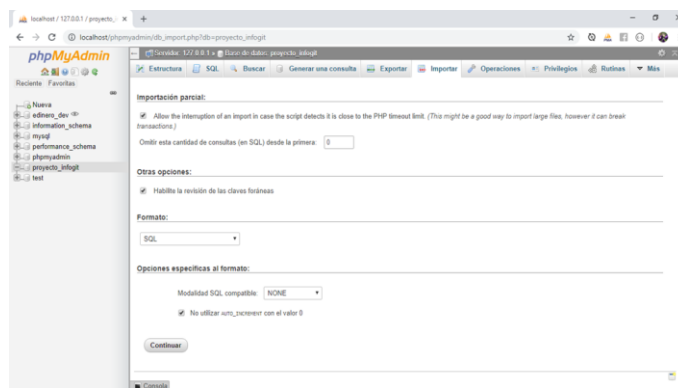


Figura 33. Continuación del proceso de importación de la base de datos.

Se comprueba la creación de las tablas necesarias con sus respectivos datos.

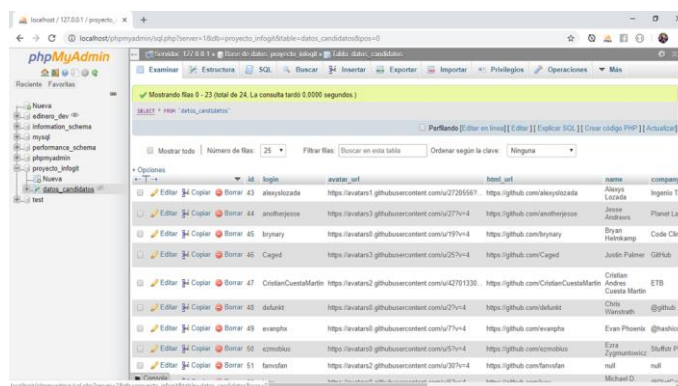


Figura 34. Comprobación de tablas de la base de datos.

Se da click en la pestaña *Estructura*, para comprobar los campos de la tabla

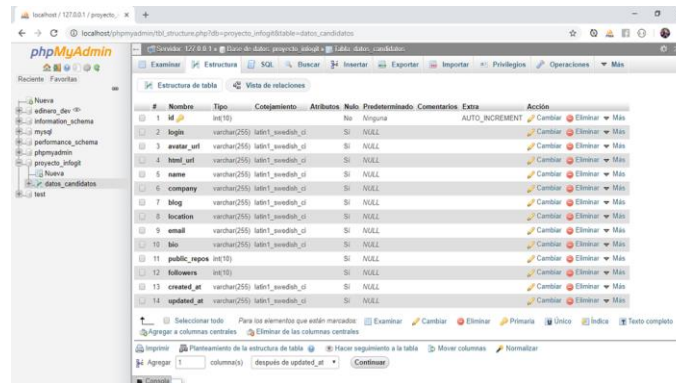


Figura 35. Comprobación de estructura de las tablas.

Se procede a dar click al botón Admin del servicio de Apache.

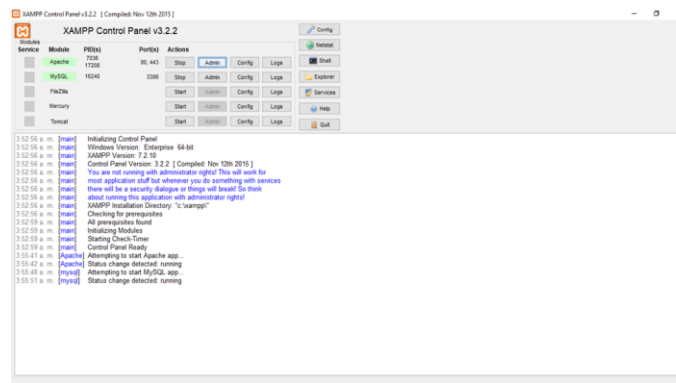


Figura 36. Botón Admin servicio apache.

Esta acción abrirá un navegador web y muestra los proyectos que tenemos dentro de la carpeta htdocs. Se procede a dar click a la carpeta proyecto

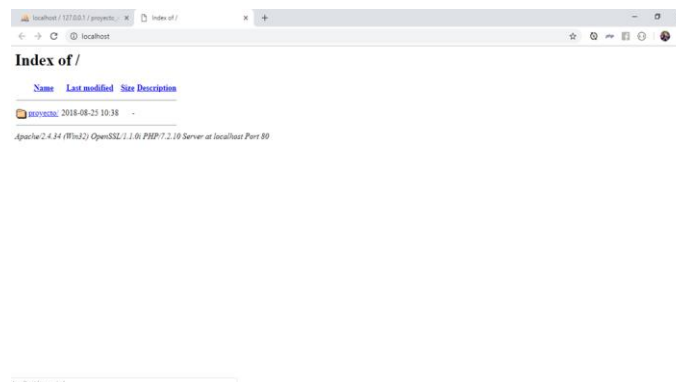


Figura 37. Carga de proyectos por medio del servicio Apache.

Se abre el archivo *index.php*, donde se observa el listado de los archivos Json almacenados en la carpeta *archivos*.



Figura 38. Carga del archivo *index.php*.

En la Fig. 38, se pueden ver los archivos Json alojados en la carpeta *archivos*.

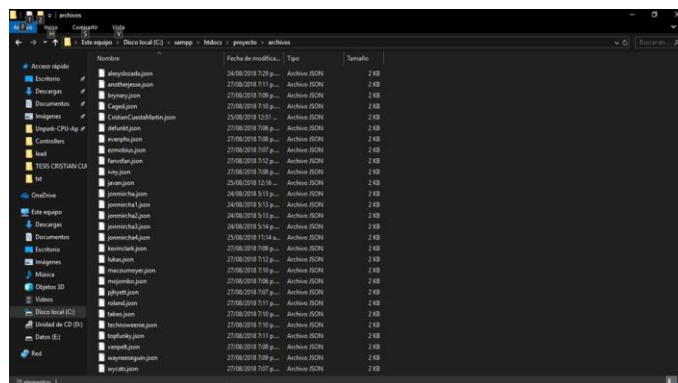


Figura 39. Archivos Json almacenados.

Se da click en el botón *ver*, de alguno de los elementos listados en la página *archivos*. Esta acción remite a la página que se observa en la Fig. 40, donde muestra la información más relevante del Json seleccionado y se elige si se agrega a la base de datos o se descarta ese candidato

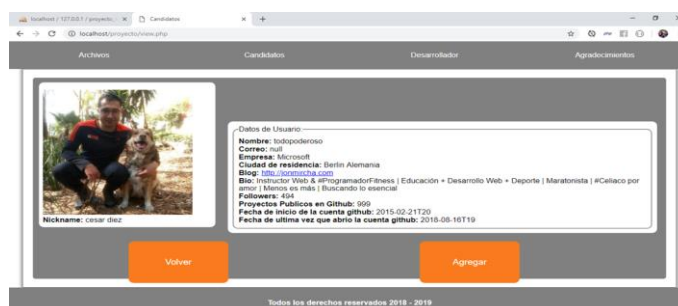


Figura 40. Información del archivo Json.

VI. RESULTADOS ADQUIRIDOS

Al dar click en la opción *Candidatos*, se muestra una lista organizada de los registros que se han guardado en la base de datos de mayor a menor por la cantidad de repositorios públicos, los tres primero serán resaltados en color verde. Esta plataforma de visualización esta diseñada para facilitar el trabajo, filtrando los parámetros que se han registrado previamente a la plataforma, con el objetivo de que los recursos sean aplicados eficientemente, mediante el análisis, control y muestra de la información.

Esta información se clasificará en las siguientes columnas, de los registros guardados en la base de datos:

- Nombre
- Email
- Lugar de residencia
- Fecha de creación de la cuenta
- Fecha de ultima conexión
- Repositorios públicos
- seguidores

Nombre	Email	Empresa	Lugar de residencia	Fecha creación de cuenta	Fecha ultima conexión	Repositorios Públicos	Seguidores
IsidoroBero	isidoro	Utopia	Central America	2018-02-21 12:00	2018-08-16 11:19	209	63
Yehuda Katz	null	Vide Inc.	San Francisco	2008-01-12 10:05	2018-08-17 11:17	209	6976
rik danger otton	null	GitHub	Louville CO	2008-01-14 10:04	2018-08-20 12:20	168	2454
Bryan Helmick	null	Code Climate	New York City	2008-01-13 11:10	2018-08-25 10:07	163	562
Evan Freener	null	@hashicorp	Los Angeles CA	2008-01-13 11:16	2018-08-13 11:18	138	1150
Justin Palmer	null	GitHub	Portland OR	2008-01-15 10:04	2018-08-24 11:16	119	1791
Chris Wanstrath	null	@github	San Francisco	2007-10-20 10:05	2018-08-15 10:02	107	20407
Jonathan MirCha	null	jonmircha	Mexico CDMX	2015-02-21 12:20	2018-08-16 11:19	100	494
Wayne E. Segun	null	hep	Buffalo NY	2008-01-13 11:06	2018-08-27 11:13	97	677
Geoffroy Grounouch	null	@hashicorp	Seattle USA	2008-01-15 10:05	2018-08-21 11:17	92	1166
Jesse Andrews	null	Planet Labs	San Francisco CA	2008-01-15 10:07	2018-07-12 11:15	88	143
Michael D. Ivey	null	@NotGames	Luscomb AL	2008-01-12 11:15	2018-08-22 11:14	86	81
Tom Preston-Werner	null	null	San Francisco	2007-10-20 10:05	2018-08-20 11:17	60	21164
Ahmad Mahmassani	null	basecamp	Ann Arbor MI	2008-04-26 10:05	2018-07-23 11:19	57	463
Marc-André Cournoyer	null	null	Montreal QC Canada	2008-01-14 10:05	2018-07-12 11:15	56	687
Chris Van Pelt	null	crowdfunder.com	San Francisco	2008-01-13 10:05	2018-07-16 11:23	45	70
Aleays Lozada	null	Ingenio T1	Bogota Colombia	2012-11-04 11:14	2018-08-24 10:01	36	103
Ezra Zygmuntowicz	null	Stufor P&C	In the NW	2008-01-12 10:07	2018-08-22 11:14	22	475
Kevin Choo	null	ojo	null	2008-05-13 11:14	2018-08-20 11:03	22	34

Figura 41. Registros guardados.

Al dar click en la opción *Desarrollador*, se mostrará la información personal del autor del proyecto.

Estos son los candidatos organizados por mayor cantidad de proyectos en github

Archivos **Candidatos** Desarrollador Agradecimientos

Nombre: Cristian Andres Cuesta Martin
 Correo: cuestamartin@gmail.com
 Universidad: Politécnica de Valencia
 Ciudad de residencia: Bogotá
 Redes Sociales: <https://www.facebook.com/cristian.cuestamartin>
 Título: Máster en Tecnologías redes y sistemas de comunicaciones
 Followers: 0
 Proyectos Públicos en GitHub: 0
 Fecha de inicio de la cuenta github: 0
 Fecha de ultima vez que abrio la cuenta github: 0

Todos los derechos reservados 2018 - 2019

Figura 42. Información del desarrollador.

Al dar click en la opción *Agradecimientos*, se mostrará la información del ingeniero Manuel Esteve Domingo.

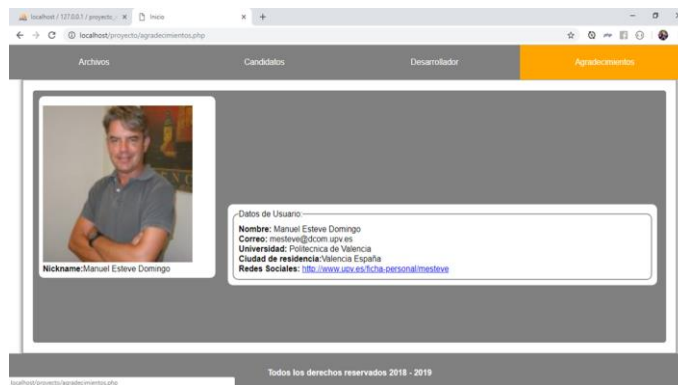


Figura 43. Información del director de proyecto.

Los Json son sacados de Github. En este caso, se tiene un ejemplo de un desarrollador conocido en el medio. La información de esta red social para programadores está abierta y basta con conocer su nickname para introducirlo en la siguiente dirección [https://api.github.com/users/\(Nombre de usuario\)](https://api.github.com/users/(Nombre de usuario))

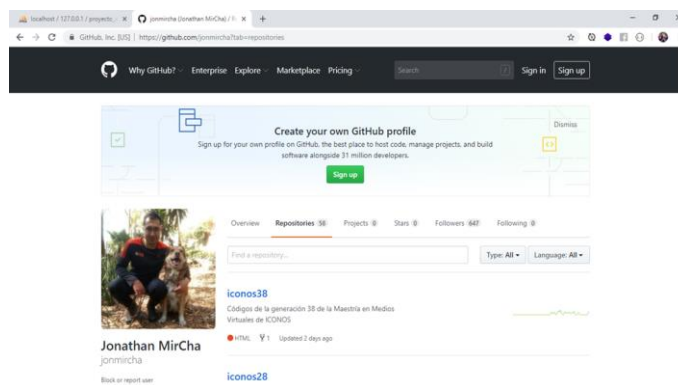


Figura 44. Información obtenida de Github.

La dirección <https://api.github.com/users/jonmircha>, retorna la información en formato Json. Se puede dar click derecho, guardarlo y posteriormente llevarlo al directorio archivo del proyecto para que sea leído automáticamente. Al dar click en *ver*, estar seguros de que es un buen candidato, dar al botón *agregar* para guardar la información de la base de datos.

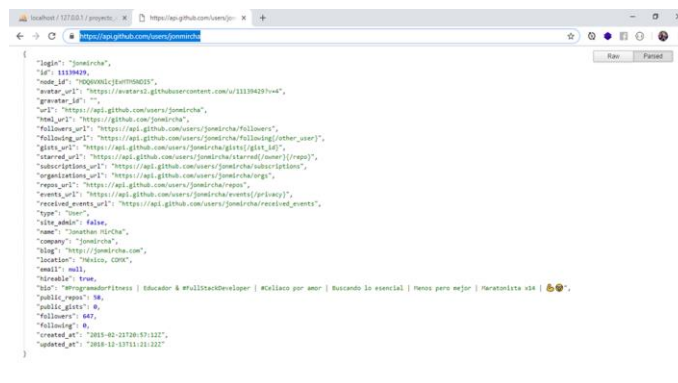


Figura 45. Información reflejada de Github.

VII. CONCLUSIONES

Se realizó el diseño y desarrollo de un aplicativo web, que se usa como herramienta de un modelo OSINT para el análisis, recolección y depuración de la información de la red social Github, para su procesamiento en una base de datos y la visualización en orden alfabético de la información, organizada en orden ascendente, seleccionando los tres mejores desarrolladores con un color verde. Esto se desarrolló en un equipo anfitrión conformado basado en sistema IOS de Apple. Esto se logró gracias a la integración en el conjunto de procesos, aplicaciones y tecnologías que facilitan la obtención rápida y sencilla de datos provenientes de los sistemas de gestión tecnológica y redes sociales como Github que es donde se capturaron los datos, para su análisis e interpretación automática de acuerdo a los filtros parametrizados con anterioridad en la plataforma, de manera que puedan ser aprovechados para la toma de decisiones de las personas y se conviertan en conocimiento base para los responsables en la toma de decisiones, de un grupo de desarrolladores que tengan mayor cantidad de repositorios, es decir más trabajos publicados en esa red social.

VIII. LINEAS FUTURAS

Partiendo del presente estudio se plantea promover la generación de desarrollo, como base para futuros análisis a medida del avance tecnológico, como las herramientas de inteligencia de negocios, en donde se ha identificado la creciente demanda del sector TIC en captar datos de diferentes fuentes, que se convierten en información tras ser procesados adecuadamente. El análisis de esta información proporciona conocimiento y el conocimiento es poder, este trabajo de fin de master es un sustento teórico para implementaciones futuras en un entorno empresarial, donde se requiera hacer análisis de los datos, como por ejemplo el comportamiento de los usuarios en redes sociales, cuáles son los lugares más visitados por las ubicaciones que comparten los usuarios. Esto conlleva un reto en temas de seguridad y gobernanza de los datos que deberán ser regulados por un marco legislativo.

IX. AGRADECIMIENTOS

Como primera instancia, agradecer a mi familia por el constante apoyo y ánimo para cumplir con este logro y el desarrollo constante de mi persona.

A mis compañeros por la experiencia compartida y las enseñanzas otorgadas.

A mi director de carrera por el conocimiento entregado, el carisma y la disciplina aplicada para lograr esta meta.

Por último, a todas las personas que de una u otra manera participaron en las fortalezas que adquirí en este camino.

X. REFERENCIAS

- [1] M. Rouse, <<MySQL (Guía Esencial: Las bases de datos dan soporte a las tendencias de TI)>>, 1 enero 2015. [En línea]. Available: <https://searchdatacenter.techtarget.com/es/definicion/MySQL>. [Último acceso: 27 junio 2019]
- [2] G. B., <<¿Qué es Apache? Descripción completa del servidor web Apache>>, 18 enero 2019. [En línea]. Available: <https://www.hostinger.co/tutoriales/que-es-apache/>. [Último acceso: 27 junio 2019]
- [3] J. Grau, <<Xampp. Herramienta indispensable para desarrollo web>>, 12 julio 2017. [En línea]. Available: <http://www.codigo-binario.es/xampp-herramienta-para-dev-web/>. [Último acceso: 27 junio 2019]
- [4] I. Romero, <<¿Qué es HTML y para qué sirve?>>. [En línea]. Available: <http://www.acercadehtml.com/manual-html/que-es-html.html>. [Último acceso: 27 junio 2019]
- [5] Masadelante.com, <<Definición de CSS - ¿Qué son las hojas de estilo o cascading style sheets?>>. [En línea]. Available: <https://www.masadelante.com/faqs/css>. [Último acceso: 27 junio 2019]
- [6] D. Pérez, <<¿Qué es Javascript?>>, 3 julio 2007. [En línea]. Available: <http://www.maestrosdelweb.com/que-es-javascript/>. [Último acceso: 27 junio 2019]
- [7] Php.net, <<¿Qué es PHP?>>. [En línea]. Available: <https://php.net/manual/es/intro-what-is.php>. [Último acceso: 27 junio 2019]
- [8] D. Guzmán, <<¿Qué es JSON?>>, 28 junio 2018. [En línea]. Available: <https://codingornot.com/que-es-json>. [Último acceso: 27 junio 2019]
- [9] K. de la Cuadra, <<¿Qué es GitHub?>>, 5 julio 2018. [En línea]. Available: <https://platzi.com/blog/que-es-github/>. [Último acceso: 27 junio 2019]
- [10] Cano, J. J. (2011). Ciberseguridad y ciberdefensa: dos tendencias emergentes en un contexto global. *ASOCIACION COLOMBIANA DE INGENIEROS DE SISTEMAS*, 119, págs. 4-7.
- [11] Department Of The Army - United States Of America. (22 de 10 de 2013). *Defense Technical Information Center*. Obtenido de www.dtic.mil: http://www.dtic.mil/doctrine/new_pubs/jp2_0.pdf
- [12] ITU. (2010). *ITU (Unión Internacional de Telecomunicaciones)*. Recuperado el 30 de 06 de 2017, de www.itu.int: <http://www.itu.int/net/itunews/issues/2010/09/20-es.aspx>
- [13] Juez, B. (15 de 03 de 2017). *Estados Unidos acusa a dos espías rusos del robo de datos a Yahoo*. Obtenido de El Mundo: <http://www.elmundo.es/internacional/2017/03/15/58c9503422601df72e8b4603.html>
- [14] Matherly, J. (s.f.). *theHarvester*. Obtenido de <https://github.com/laramies/theHarvester>
- [15] Melo, P. (13 de Nov de 2014). *High Availability by LOGITEK*. (P. Melo, Ed.) Obtenido de www.altadisponibilidadlogitek.com: <http://www.altadisponibilidadlogitek.com/la-disponibilidad-como-el-primero-de-los-3-ejes-fundamentales-de-la-ciberseguridad-industrial/>

- [16] Pérez, I. (19 de Febrero de 2014). *Web Live Security*. Obtenido de <https://www.welivesecurity.com/la-es/2014/02/19/maltego-herramienta-muestra-tan-expuesto-estas-internet/>
- [17] Sánchez, J. (12 de 05 de 2017). *ABC*. Recuperado el 30 de 06 de 2017, de <http://www.abc.es/>: http://www.abc.es/tecnologia/redes/abci-telefonica-sufre-ciberataque-interna-201705121304_noticia.html