Data Lake - Visualización de datos

Anderson Iván Cordova Calvopiña
Ronny Patricio Cajas Benitez
Vicente David Delgado Guiz
anderson.cordova@epn.edu.ec
ronny.cajas@epn.edu.ec
vicente.delgado@epn.edu.ec

Resumen – Este documento tiene por objetivo explicar el procedimiento llevado a cabo en la alimentación de un Data Lake y creación de Dashboards, para su posterior análisis y obtención de visualizaciones con carácter relevante en 5 distintas temáticas. Utilizando tecnologías de web scraping, recolección de datos y fuentes oficiales.

Durante el proceso se cubrirán las etapas de cosecha, limpieza y análisis de los datos; se incluye un mapping y su traslado a un clúster que se usará como intermedio con el software de visualización.

Se pretende obtener una información que resulte útil para cualquier usuario que se interese por cualquiera de las temáticas abordadas.

I. INTRODUCCIÓN

La información es el bien no tangible más valioso en las últimas décadas, por ende, se ha vuelto una necesidad el poder procesarla, analizarla y utilizarla a conveniencia en distintos campos de la sociedad. Como consecuencia se han obtenido lugares de almacenamientos masivos de datos brutos, como fuentes centralizadas de las cuales recabar información valiosa, estos lugares son conocidos como Data Lake's.
Un Data Lake se han convertido en una forma de describir cualquier exponencial conjunto de datos en el que la arquitectura y los requisitos de los mismos no se definen hasta que son consultados. La información contenida puede normalizarse y enriquecerse, incluyendo extracción de metadatos, trasformación de formatos, mutaciones, aumentos, extracción de entidades, reticulaciónes, agregaciones, des-normalización o indexación. Siendo por consecuencia la fuente idónea para visualizaciones pertenecientes de un Dashboard **[1]**.

Un dashboard es una herramienta de gestión de la información que nos permite monitorizar, analizar y mostrar de manera dinámica los indicadores clave de desempeño, métricas y controles para hacer un seguimiento del estado de cualquier temática de interés, desde el análisis de un negocio hasta las tendencias en redes sociales. Resulta un centro de control de información o también podemos considerarlo como un resumen de la información personalizado, con presentaciones visuales de las métricas, con enfoque práctico y en varias ocasiones procesando dicha información en tiempo real **[2]**.

.

II. DEFINICIÓN DEL CASO DE ESTUDIO

Se requiere obtener el comportamiento de usuarios en la red, obtenido 5 dashboards de las temáticas nombradas a continuación:

1. Pulso político de Ecuador por provincias y ciudades principales.
2. Juegos en línea por países.
3. Música en tendencia.
4. Afección del Covid-19 a nivel mundial.
5. Manifestaciones en Ecuador, septiembre de 2020

III. OBJETIVOS GENERALES Y ESPECÍFICOS

- *Objetivos Generales*
  - Almacenar la data de todas las temáticas en un clúster con su correspondiente background.
  - Generar un dasboard por cada temática abordada.
  - Generar una arquitectura estable y robusta para alimentar las visualizaciones correspondientes.
- *Objetivos Específicos*
  - Conceptualizar la actividad política de los usuarios en Ecuador.
  - Encontrar la actividad de los usuarios con los videojuegos entre el 2013 al 2020.
  - Describir las tendencias musicales en la red.
  - Conceptualizar el comportamiento de internautas a nivel mundial frente al Covid-19
  - Recabar los últimos acontecimientos de las manifestaciones en Ecuador

IV. DESCRIPCIÓN DEL EQUIPO DE TRABAJO Y ACTIVIDADES REALIZADAS POR CADA MIEMBRO DEL EQUIPO.

Para el desarrollo del proyecto se cuenta con la capacidad de 3 estudiantes actualmente cruzando la carrera en Tecnología Superior de Desarrollo de Software en la Escuela Politécnica Nacional del Ecuador, autores de este informe, y sus correspondientes equipos ofimáticos.

Cada integrante se encarga de las siguientes actividades:

- *Cosecha:* se recopila información de las temáticas a través de web scraping, recopilación de twitter, y fuentes oficiales.
- *Transformación de datos:* se debe tratar los datos obtenidos para entender su contenido.
- *Mapping:* se deben mapear los datos obtenidos para poder procesarlos posteriormente.

- *Replicación:* en este punto los 3 miembro del equipo compartirán su trabajo, para crear un solo punto de partida hacia el clúster y posteriores visualizaciones.
- *Transferencia al clúster:* los datos mapeados deben ser indexados al clúster.
- *Creación de dasboards:* se utiliza los datos obtenidos a través de software de visualizaciones para generar los correspondientes dashboards.

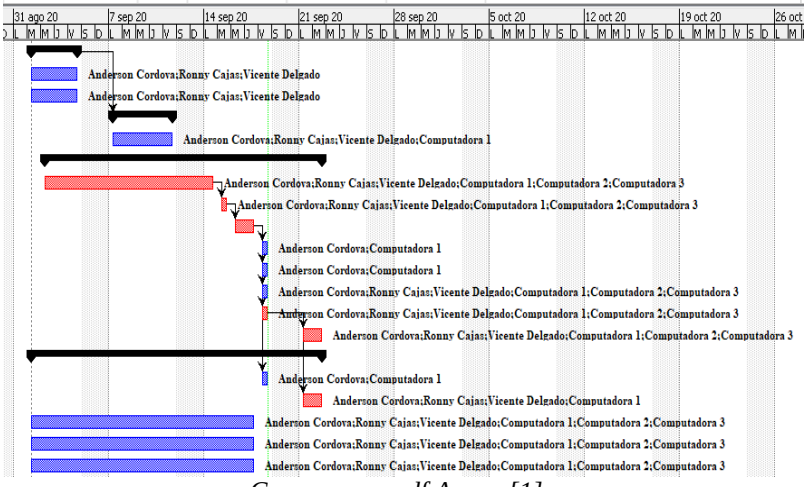
Nota: Un integrante trabajara en 2 bases de datos SQL y luego migrara a NoSQL, mientras los 2 miembros empezaran el proceso en bases NoSQL. Esto se define por preferencia del equipo.

Nota: Los temas son sorteados entre los miembros del equipo para cubrir con todas las actividades

V. CRONOGRAMA DE ACTIVIDADES.

Con el siguiente cronograma definimos plazos para cada actividad a nivel general del proyecto,

Nombre	Duracion	Inicio	Terminado	Predecesores	Nombres del Recurso
Analisis	4 days	1/09/20 8:00	4/09/20 17:00		
Definir tecnologias	4 days	1/09/20 8:00	4/09/20 17:00		Anderson Cordova;Ronny Cajas;Vicente Delgado
Definir metodos	4 days	1/09/20 8:00	4/09/20 17:00		Anderson Cordova;Ronny Cajas;Vicente Delgado
Diseño	5 days	7/09/20 8:00	11/09/20 17:00	1	
Diseño de arquitectura Data_Lake	5 days	7/09/20 8:00	11/09/20 17:00		Anderson Cordova;Ronny Cajas;Vicente Delgado;Computadora 1
Implementacion	15 days	2/09/20 8:00	22/09/20 17:00		
Cosecha	9 days	2/09/20 8:00	14/09/20 17:00		Anderson Cordova;Ronny Cajas;Vicente Delgado;Computadora 1;Computadora 2;Computadora 3
Transformación de datos	1 day	15/09/20 8:00	15/09/20 17:00	7	Anderson Cordova;Ronny Cajas;Vicente Delgado;Computadora 1;Computadora 2;Computadora 3
Replicacion de datos	2 days	16/09/20 8:00	17/09/20 17:00	8	
Creación de índices	1 day	18/09/20 8:00	18/09/20 17:00	9	Anderson Cordova;Computadora 1
Creación de mapping	1 day	18/09/20 8:00	18/09/20 17:00	9	Anderson Cordova;Computadora 1
Traferencia a elasticsearch	1 day	18/09/20 8:00	18/09/20 17:00	9	Anderson Cordova;Ronny Cajas;Vicente Delgado;Computadora 1;Computadora 2;Computadora 3
Indexar datos a kibana	1 day	18/09/20 8:00	18/09/20 17:00	9	Anderson Cordova;Ronny Cajas;Vicente Delgado;Computadora 1;Computadora 2;Computadora 3
Creacion de Dasboards	2 days	21/09/20 8:00	22/09/20 17:00	13	Anderson Cordova;Ronny Cajas;Vicente Delgado;Computadora 1;Computadora 2;Computadora 3
Documentacion	16 days	1/09/20 8:00	22/09/20 17:00		
Creacion de archivo contenedor	1 day	18/09/20 8:00	18/09/20 17:00	9	Anderson Cordova;Computadora 1
Creacion y publicacion de video explic	2 days	21/09/20 8:00	22/09/20 17:00	13	Anderson Cordova;Ronny Cajas;Vicente Delgado;Computadora 1
Realizacion de informe	13 days	1/09/20 8:00	17/09/20 17:00		Anderson Cordova;Ronny Cajas;Vicente Delgado;Computadora 1;Computadora 2;Computadora 3
Realizacion de presentacion	13 days	1/09/20 8:00	17/09/20 17:00		Anderson Cordova;Ronny Cajas;Vicente Delgado;Computadora 1;Computadora 2;Computadora 3
Repositoio git	13 days	1/09/20 8:00	17/09/20 17:00		Anderson Cordova;Ronny Cajas;Vicente Delgado;Computadora 1;Computadora 2;Computadora 3



Cronograma.pdf Anexo [1]

VI. RECURSO Y HERRAMIENTAS UTILIZADAS.

- *PROJECTLIBRE*

El software gratuito ideal para el diseño de cronogramas profesionales o semi-profesionales en cuanto a proyectos **[3]**.

Es un software de código abierto para la administración de proyectos, que se ejecuta sobre la plataforma de Java que proporciona una interfaz de usuario bastante intuitiva, sencilla de utilizar, ofrece bastantes funcionalidades muy completas para la definición de actividades en cuanto al tiempo, al igual que la integración de los recursos en cada actividad **[3]**.

ProjectLibre Referencia [3]

- PYTHON

Python es un lenguaje de programación con propósito general, por lo que puede desarrollarse en distintas áreas de la información. Es un lenguaje interactivo e interpretado con funciones incorporadas para el tratamiento de sus variables y cuenta con una sintaxis clara y fácil de entender **[4]**.

Para este proyecto en específico se está haciendo uso de las siguientes librerías:

- *pymongo:* utilizada para poder conectarnos a MongoDB.
- *json:* principalmente utilizada para parsear el JSON de archivos o string.
- *tweepy:* librería fácil de usar para acceder a la API de Twitter.
- *couchdb:* utilizada para poder conectarnos a CouchDB.
- *requests:* facilita la abstracción del proceso de hacer solicitudes HTTP
- *argparse:* para construir procesadores de argumentos y opciones de línea de comando.
- *postgres:* utilizada para poder conectarnos a PostgreSQL.
- *pymysql:* utilizada para poder conectarnos a MySQL
- *pandas:* es una biblioteca escrita como extensión de NumPy para manipulación y análisis de datos.

Python Referencia [4]

- *APACHE COUCHDB*

El gestor de base de datos de código abierto Apache CouchDB o simplemente llamada CouchDB es una base NoSQL con el foco puesto para simular la web, utiliza Json para almacenar sus datos, Javascript como el lenguaje para realizar consultas sobre la base por medios de API's **[5]**.

CouchDB no almacena los datos y sus relaciones en tablas. En cambio, cada base de datos es una colección de documentos independientes. Cada documento mantiene sus propios datos y su esquema auto-contenido **[5]**.

El protocolo que emplea CouchDB en su replicación de datos se implementa en una variedad de proyectos y productos que abarcan todos los entornos informáticos desde clústeres de servidores distribuidos globalmente, pasando por teléfonos móviles hasta navegadores web **[5]**

CouchDB Referencia [5]

- *MONGODB*

MongoDB es una base de datos libre distribuida, basada en documentos y con aplicación general que ha sido diseñada para desarrolladores de aplicaciones modernas por su rango de escalabilidad y con tecnología acorde a la era de la nube **[6]**.

El modelo para trabajar con los documentos en MongoDB resulta ituitivo y fácil de aprender ya que proporciona a los desarrolladores todas las funcionalidades que necesitan para satisfacer los requisitos complejos a cualquier escala. Se proporcionan drivers para más de diez lenguajes, y además es sostenida por la comunidad desarrolladora **[6]**.

MongoDB Referencia [6]

- *POSTGRESQL*

Es una base de datos distribuida de codigo abierto de tipo relacional, aunque es su arquitectura es posible ejecutar búsquedas no relacionales **[7]**.

Dentro del manejo de sus documentos posee data types avanzados que permiten la facilidad de ejecutar optimizaciones de rendimiento y trabajan con Json's **[7].**

PostgreSQL Referencia [7]

- *XAMPP*

XAMPP es una distribución de Apache gratuita y de fácil instalación que contiene MariaDB, PHP y Perl. El paquete de instalación de XAMPP ha sido diseñado para que resulte optimo y fácil de instalar y usar **[8]**.

Una gran ventaja que nos ofrece es en cuanto a tiempo y recursos en instalar y ejecución, pues con XAMPP los componentes no actúan por separado, sino que sólo se requiere una pequeña fracción del tiempo necesario para descargar y ejecutar un archivo ZIP, tar, exe o fkl **[8].**

XAMPP Referencia [8]

- *ELK*

Se describen los componentes de ELK a continuación:

- *LOGSTASH*

Este componente es el paso a elasticsearch, y corresponde a la parte del procesamiento para ingresar nuestros datos que pueden ser en una multitud de fuentes, transformándolos y enviándolo a su destino mediante clases input y output respectivamente **[9]**.

logstash Referencia [9]

- *ELASTICSEARCH*

Motor de búsqueda y analítica, que funciona como base de datos distribuida tanto a nivel de procesamiento como de información, otorgando consultas con mejores rendimientos **[10].**

Elasticsearch Referencia [10]

- *KIBANA*

Es el componente más visual de la ELK, en la que se puede manipular los datos a fin de conseguir visualizaciones en distintos tipos, comprender como fluyen las solicitudes en servicios y generación dashboards **[11]**.

kiban Referencia [11]

- *TABLEAU PUBLIC*

Es un software de visualización de datos gratuito que a partir de fuentes que puedes ser entre archivos de Excel como CSV u otros orígenes de datos, permiten generar visualizaciones de alto impacto gráfico e interactivo **[12]**.

Además, una gran ventaja es que con la instalación de la herramienta se podrá crear libremente un perfil personal en la nube con 10 GB de espacio en donde podrás publicar tus visualizaciones y compartirlas de manera más eficiente y profesional **[12]**.

TableauPublic Referencia [12]

- *POWER BI*

Power BI es una herramienta software de Microsoft, para el análisis de datos en el ámbito empresarial, que ofrece un alto rendimiento en visualizaciones de resultados, por sus graficas dinámicas e interfaz fácil de usar. Power BI incorpora capacidades de inteligencia empresarial, con el que facilita el entendimiento de gráficos y visualizaciones, para crear informes y paneles. Power BI ofrece también protección de extremo a extremo en la transmisión de datos, conexión a servicios de nube y otros servicios de Microsoft

Power BI Referencia **[13]**

- *Hamachi*

Hamachi es un software que simula una red local entre 2 o mas equipos. Fue desarrollado por LogMeIn, este software permite la conexión de quipos remots para intercambiar archivos entre diferentes máquinas conectadas de una manera rápida y sencilla

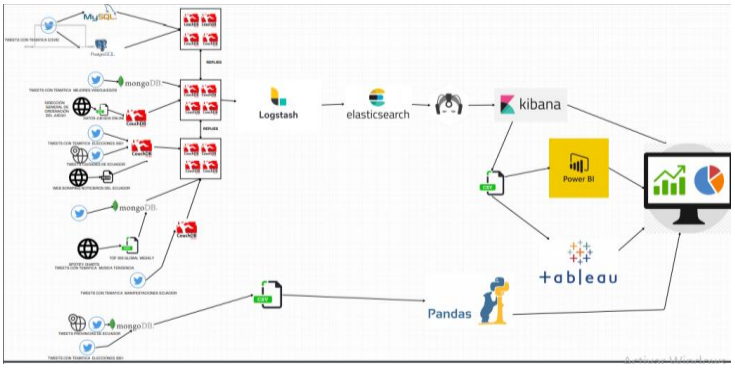
VII. ARQUITECTURA DE LA SOLUCIÓN.

En nuestra arquitectura se plantea una recopilación masiva de datos a traves de diversas fuentes en las que incluimos sondeos de web scrapping, recopilación de la red social twitter mediante el uso de API's y obtención de csv's en fuentes oficiales relacionadas con los temas a tratar.

Tras la cosecha, se trasladarán todos los datos en distintas bases de datos en couchDB, en donde los miembros del equipo tendrán la posibilidad de replicar el trabajo, creando una arquitectura más robusta.

Una vez realizado el mapping de estas bases se procederá a utilizar ELK para generar las visualizaciones trasladando los datos por todos los componentes, como interfaz gráfica utilizaremos cerebro para conectarnos a elasticsearch y controlar el proceso.

Finalmente se generarán los dashboards correspondientes, en este punto, se puede trasladar la información en tableau public para pulir las visualizaciones o generar nuevas, según sea necesario.



Data-Lake.pdf Anexo [1]

VIII. EXTRACCIÓN DE DATOS.

- *API Twitter*

Se generan scripts .py para recopilar acerca de la temática de elecciones 2021, de música en tendencia, Covid-19, mejores videojuegos. Bajo este mismo concepto se realizan scripts en base a datos geográficos, cubriendo con las 24 provincias y 20 ciudades principales del Ecuador, que serán alimentados con API's de la red social Twitter.

```python
class listener(StreamListener):

    def on_data(self, data):
        dictTweet = json.loads(data)
        try:
            dictTweet["_id"] = str(dictTweet['id'])

            doc = tweets.insert_one(dictTweet);
            print ("SAVED" + str(doc) +"=>" + str(data))

        except:
            print ("Already exists")
            pass
        return True

    def on_error(self, status):
        print (status)

auth = OAuthHandler(ckey, csecret)
auth.set_access_token(atoken, asecret)
twitterStream = Stream(auth, listener())


'''==============WORDS=============='''
words=['#Ecuador','#EcuadorEnCrisis','#universidades','#
'#LaEducacionPublicaSeDefiende','#SinCienciaNoHayFuturo',
'#PorLaSaludYEducacion','#NoAlRecortePrespuestario','#La
'Richard Martinez','Romo']
twitterStream.filter(track=words)
```

tweets_mongodb_listenere.py Anexo [1]

```python
server = couchdb.Server('http://admin:ander123@localhost:5984/')
try:
    db = server.create('politico')
except:
    db = server['politico']

'''==============LOCATIONS 20 CITIES ECUADOR============='''
twitterStream.filter(locations=[-78.619545,-0.365889,-78.441315,-0.047208])#Quito
#twitterStream.filter(locations=[-79.95912,-2.287573,-79.856351,-2.053362])#Guayaquil
#twitterStream.filter(locations=[-79.5983,-3.1761,-78.8471,-2.5578]) #Cuenca
#twitterStream.filter(locations=[-79.5484,-0.6987,-78.7461,0.0191]) #Santo Domingo
#twitterStream.filter(locations=[-80.02869,-3.354746,-79.84235,-3.190935])#Machala
#twitterStream.filter(locations=[-80.912795,-1.135691,-80.663985,-0.928689])#Manta
#twitterStream.filter(locations=[-80.5625,-1.202987,-80.316762,-0.929944]) #Potoviejo
#twitterStream.filter(locations=[-79.269946,-4.132863,-79.137379,-3.850106])#Loja
#twitterStream.filter(locations=[-78.937982,-1.470916,-78.5368,-1.1098])#Ambato
#twitterStream.filter(locations=[-79.8353,0.5667,-79.4046,1.0486]) #Esmeraldas
#twitterStream.filter(locations=[-79.616489,-1.194387,-79.368954,-0.939314]) #Quevedo
#twitterStream.filter(locations=[-78.479556,-0.412589,-78.393941,-0.290761]) #Sangolqui
#twitterStream.filter(locations=[-78.723592,-1.710588,-78.595487,-1.633713]) #Riobamba
#twitterStream.filter(locations=[-78.17586,0.260163,-78.025016,0.499894]) #Ibarra
#twitterStream.filter(locations=[-79.671471,-2.133047,-79.201259,-1.619683]) #Babahoyo
#twitterStream.filter(locations=[-78.667182,-1.008675,-78.40152,-0.867442]) #Latacunga
#twitterStream.filter(locations=[-79.893517,-2.167096,-79.6477,-1.863685]) #Samborondon
#twitterStream.filter(locations=[-80.245823,-3.502263,-80.117762,-3.416315]) #Huaquillas
#twitterStream.filter(locations=[-78.5494,0.6053,-77.5255,1.1979]) #Tulcan
#twitterStream.filter(locations=[-80.2864,-0.8627,-79.5972,0.0894]) #Chone
```

tweets_couchdb_listenerg.py Anexo [1]

- *Web Scraping*

Se creó un proyecto scrapy bajo el nombre de político, donde se recopilaros los titulares de todas las páginas web oficiales de noticieros del Ecuador

```python
import scrapy
from scrapy import Spider
from scrapy import Selector

from politico.items import PoliticoItem


class PoliticoSpider(scrapy.Spider):
    name = 'politico'
    allowed_domains = ['elcomercio.com','ecuavisa.com','ecuavisa.com','rts.com.ec','tctelevision.com','el
    start_urls = [
    'https://www.elcomercio.com/actualidad',
    'https://www.elcomercio.com/data',
    'https://www.elcomercio.com/tendencias',
    'https://www.ecuavisa.com/noticias/nacional',
    'https://www.ecuavisa.com/noticias/politica',
    'https://www.ecuavisa.com/historico/noticias/politica',
    'http://www.rts.com.ec/noticias/actualidad-1',
    'https://www.tctelevision.com/noticias',
    'https://www.tctelevision.com/politica',
    'https://www.eltelegrafo.com.ec/contenido/categoria/1/politica',
    'http://www.teleamazonas.com/noticiero-24-horas/noticias-nacionales'
    ]
    #Add all url pages
    for i in range(1,100):
        start_urls.append('https://www.eluniverso.com/politica?page='+str(i))
        start_urls.append('https://www.ecuavisa.com/historico/noticias/nacional?page='+str(i))
        start_urls.append('https://www.ecuavisa.com/historico/noticias/politica?page='+str(i))
        start_urls.append('https://www.tctelevision.com/noticias/page/'+str(i))
        start_urls.append('https://www.tctelevision.com/politica/page/'+str(i))
        k=i*12
        start_urls.append('https://www.eltelegrafo.com.ec/contenido/categoria/1/politica?start='+str(k))

    def parse(self, response):
        item=PoliticoItem()
        headlines=Selector(response).xpath('//div[@class="article articulo-grande"]')
        for headline in headlines:
            item['message']=headline.xpath('a/text()').extract()[0]
            item['url']=headline.xpath('a/@href').extract()[0]
            yield item
        headlines=Selector(response).xpath('//div[@class="views-field views-field-title"]')
        for headline in headlines:
```

politico_spiders.py Anexo [1]

- *CSV's*

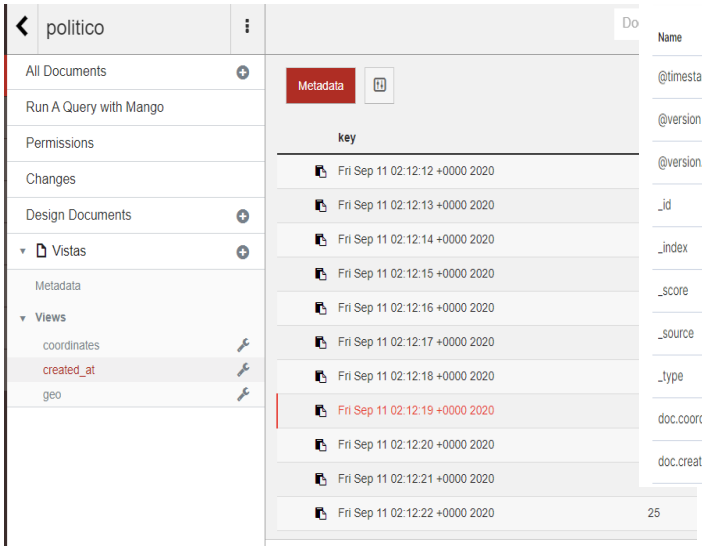
Se obtienen CSV's de los mejores juegos entre los años 2013 al 2020 de la página de la dirección general de ordenación de juegos. De igual manera se obtuvieron CSV's del Top200 de Spotify de las 10 últimas semanas a través de la web oficial de Spotify Charts.

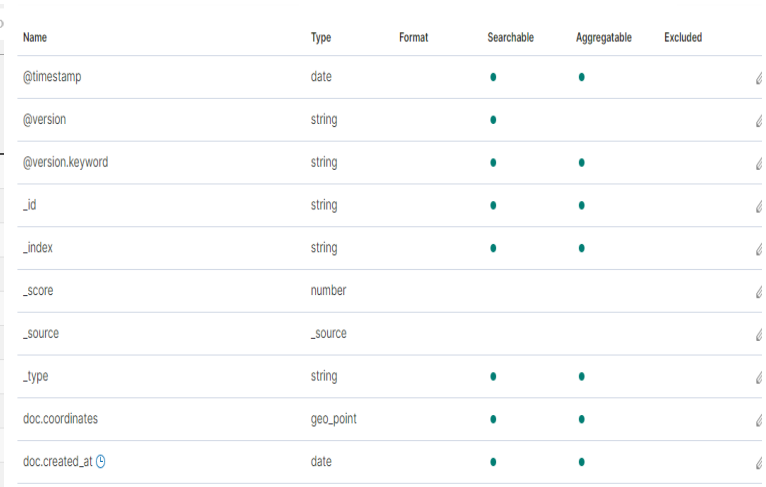
	A	B	C	D	E	F	G	H
1	Año	Trimestre	Mes	Ap. Dep. de	Ap. Dep. de	Ap. Hípicas d	Otras Ap. de	Ap. Deportiv
2	2013	2013.T1	Enero	162047262	0	1283698	0	25
3	2013	2013.T1	Febrero	157694814	0	1021289	599	13739
4	2013	2013.T1	Marzo	172671200	0	1542232	154	62887
5	2013	2013.T2	Abril	159740095	0	1567041	5970	57470
6	2013	2013.T2	Mayo	154806640	0	1864772	388	47501
7	2013	2013.T2	Junio	152493062	0	2173177	454	40474
8	2013	2013.T3	Julio	126943251	0	2029753	0	22772
9	2013	2013.T3	Agosto	164302106	0	2104109	0	30244
10	2013	2013.T3	Septiembre	183000088	0	1738763	2	35733
11	2013	2013.T4	Octubre	196785593	0	2250930	15	54780
12	2013	2013.T4	Noviembre	177819239	0	1800667	0	52868
13	2013	2013.T4	Diciembre	179906795	0	1818361	134	43087
14	2014	2014.T1	Enero	211500799	0	1948905	621	37320
15	2014	2014.T1	Febrero	197214054	0	1696579	746	39289
16	2014	2014.T1	Marzo	227826050	0	2266131	782	46569
17	2014	2014.T2	Abril	216217412	0	2512661	0	18519
18	2014	2014.T2	Mayo	222996687	0	2854476	803	18773
19	2014	2014.T2	Junio	223314871	0	3682915	0	27250
20	2014	2014.T3	Julio	206144156	0	3773802	1411	6152
21	2014	2014.T3	Agosto	227151481	0	3290734	14987	15653
22	2014	2014.T3	Septiembre	265479403	0	3024780	7183	29047
23	2014	2014.T4	Octubre	273595488	0	3165701	15547	27974

DATOS_JUEGO_ONLINE_2020.T1_CANTIDADES_JUGADAS.csv Anexo [1]

IX. ANÁLISIS DE INFORMACIÓN.

Con la información ya colocada dentro de nuestras bases en CocuhDB, se realizaron MapReduce de las bases de forma simple, para empezar a familiarizarnos con los campos y distintos componentes de nuestros datos recolectados, que posteriormente nos servirán para poder realizar los mapping en el traslado al clúster elasticsearch.





Name	Type	Format	Searchable	Aggregatable	Excluded
@timestamp	date		●	●	
@version	string		●		
@version.keyword	string		●	●	
_id	string		●	●	
_index	string		●	●	
_score	number				
_source	_source				
_type	string		●	●	
doc.coordinates	geo_point		●	●	
doc.created_at	date		●	●	

Views_couchdb_localhost

Con la pauta ya establecida y una información que ya puede ser entendida por los usuarios creamos archivos de configuración con el mapping para utilizar los campos de manera idónea en elasticsearch y kibana posteriormente.

```
{
  "mappings": {
    "properties":{
      "doc":{
        "properties": {
          "created_at": {
            "type": "date",
            "format": "EE MMM d HH:mm:ss Z yyyy||dd/MM/yyyy||dd-MM-yyyy||date_optional_time
          },
          "location": {
            "type": "geo_point"
          },
          "geo": {
           "type": "geo_point"
          },
          "coordinates": {
           "type": "geo_point"
          },
          "place": {
           "properties": {
            "coordinates":{
              "type":"geo_point"
            }
           }
          },
          "message": {
            "type": "text",
            "fields": {
              "keyword": {
                "type": "keyword",
                "ignore_above": 256
              }
            }
          }
        }
```

mapping.json Anexo [1]

X. VISUALIZACIÓN DE INFORMACIÓN.

El tratamiento de la información en este sentido, tenía que ser apartada en los patrones de índices que nos ofrece kibana, para realizar las visualizaciones. Por lo tanto, tras haber realizado el mapping correctamente incluimos un campo de control en función a la fecha de creación del documento

Como podemos darnos cuenta la trasformación de nuestros datos se ejecutó de forma correcta. Por ende, la visualización de la información que obtengamos tendrán parámetros ya específicos y de control que podremos manejar, analizar y publicar.

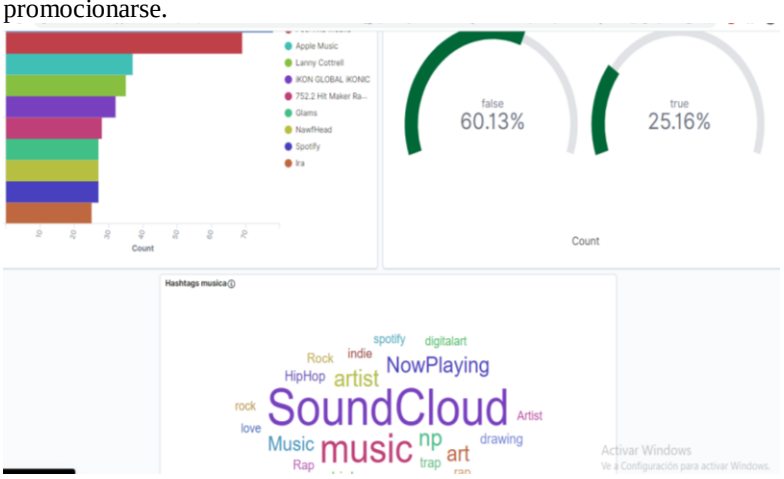
XI. RESULTADOS OBTENIDOS.

Al final del proyecto se realizaron 5 dahboards, 1 de ellos colaborativo entre kibana y tableau public para una mayor comprensión del tema.

A continuación, daremos algunos detalles de los mismos:

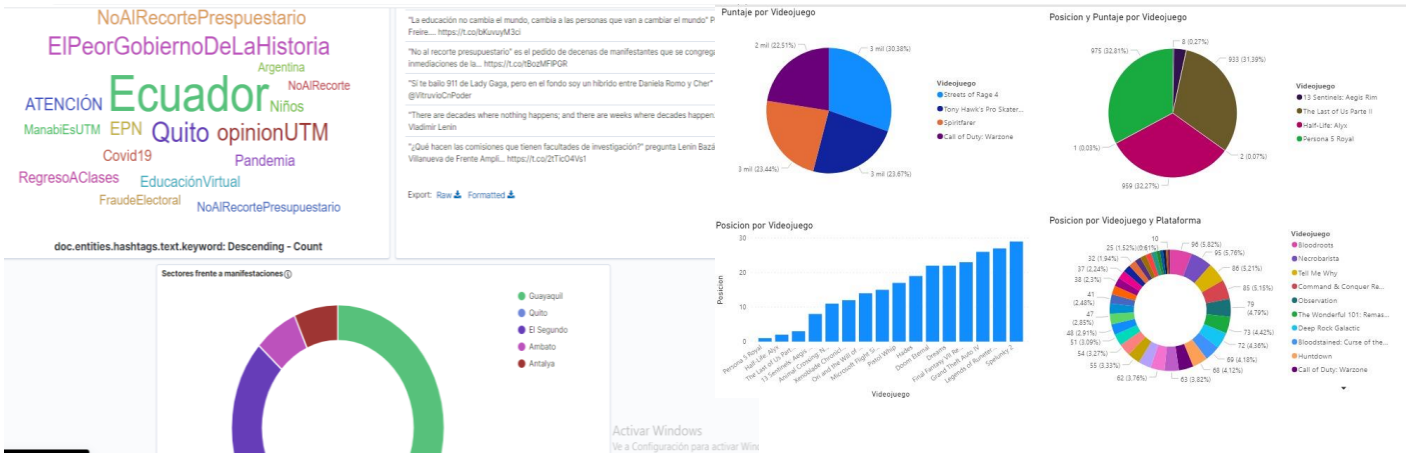
- Música

Se pudo graficar las zonas geográficas con mas interacción activa en redes sociales involucrado con ámbitos musicales, así como el comportamiento de usuarios con sus publicaciones en este tema. De igual forma se pudo establecer las compañías de música que haces mayor uso de redes sociales para promocionarse.



- Manifestaciones

Se logró recabar información de usuarios y publicaciones hechas en las jornadas de protestas en Ecuador efectuadas en septiembre del año vigente, revelando la presión en redes sociales que se efectuaba contra el gobierno.

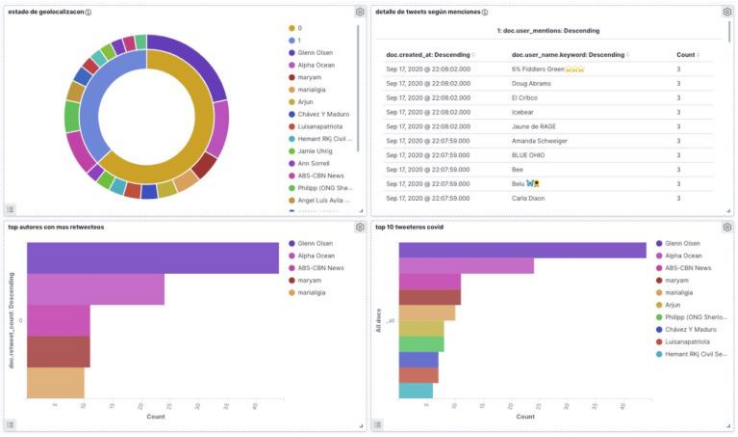


- Pulso Político

Se pudo establecer algunos comportamientos políticos en redes sociales por usuarios en Ecuador frente a las futuras elecciones 2021 y la forma en la que se estaban pronunciando al respecto, sectorizando en ciudades y provincias principales para entender el impacto que cada región genera sobre esta cuestión.

- Covid

Se pudo ver la tendencia entre países y usuarios frente a la pandemia global que nos afectó desde febrero de este año y como sigue influyendo en campaña a través de redes sociales, y en qué manera se están pronunciando los internautas sobre el tema actualmente



- Juegos

Se estableció un cubrimiento completo de la tendencia gamer de internautas en la red y su interacción sobre las redes sociales sobre la temática de los juegos, al igual que la evolución en preferencias y comportamiento entre juegos y jugadores

XII. CONCLUSIONES Y RECOMENDACIONES.

Recomendaciones:

- Antes de cosechar los datos es necesario contar con una cuenta developer de Twitter para conseguir el api que nos ayudaran en la recolección de los tweets.
- Al trabajar con basta cantidad de datos se recomienda utilizar software que soporte esta cantidad de data de manera eficiente, de igual manera contar con hardware que soporte las especificaciones de dicho software
- Es necesario asegurarse de que los datos sean limpiados y mapeados de forma correcta para conseguir visualizaciones reales.
- La etapa de cosecha para la recolección se le debe otorgar su correspondiente tiempo y con anticipación para cubrir con los datos masivos sin retrasar las siguientes actividades.

Conclusiones:

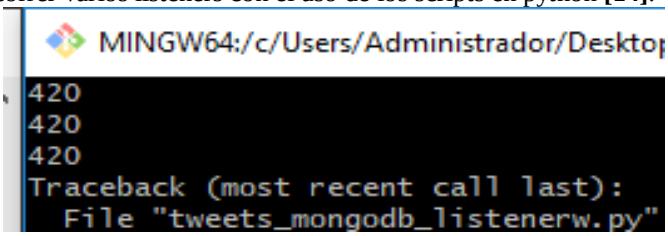
- La actividad de partidas online ha tenido una caída en los 2 últimos trimestres del 2020.
- Al analizar, se encontró que a pesar de que los datos contengan la ubicación de donde se recepta la información, la mayoría de usuario prefiere no activar la geolocalización de sus dispositivos.
- Las tendencias en redes sociales pueden ser efímeras como en el caso de las manifestaciones en septiembre, o duraderas como lo es en el caso del covid, que sigue siendo top 100 desde los inicios del año.
- Los depósitos y retiradas de video juegos han tenido gran repercusión por la pandemia de covid en el año actual.
- Los posts más retweeteados en cuanto a muestras en covid pertenecen al doctor Glenn Olsen. Reconocido especialista estadounidense.
- El número de concursos de video juegos en el 2020 ha bajado en comparación con 2013 8 millones
- YouTube, a pesar de no ser una plataforma exclusivamente de música, resulta ser el principal medio para compartirla en redes sociales

XIII. DESAFÍOS Y PROBLEMAS ENCONTRADOS.

- Error 420

En la fase de cosecha de datos, nos encontramos en repetidas ocasiones con el problema 420 por el uso de la API e Twitter indicándonos que nuestra API tiene una tarifa limitada por

realizar demasiadas solicitudes. Esto principalmente pasaba al correr varios listeners con el uso de los scripts en python **[14]**.



Error 420 localhost

- *Migración de mongoDB a couchDB*

Se encontraron conflictos con el tipo de datos que maneja mongoDb, al contener objetos anidados en nuestros datos y conflictos con el $oid que maneja couchDB, como medidas se intentó migrar a otro método en Python para compartir los datos a través de argumentos, tras unas pruebas aparentemente favorables se intentó optar por el cambio, sin embargo, al momento de pasar todas las bases existieron fallos al igual que lo tuvo posteriormente en la argumentación.

```
    raise e
# create database in CouchDB for the first time if it doesn't exi
try:
    response = requests.Session().get(URL+'/'+db+'-'+col, auth=('
    if 'error' in response.json():  # db doesnt exist on destinat
        print('Creating a new database {} in CouchDB'.format(db+'
        output = requests.Session().put(URL+'/'+db+'-'+col, auth=
        if output.status_code == 201 or 202:
            DB_COUNT += 1
    # insert documents in couchDB database.
    output = requests.Session().post(URL+'/'+db+'-'+col, auth=('a
    if output.status_code == 201 or 202:
        DOCUMENT_COUNT += 1
except requests.exceptions.RequestException as e:
```

mongo_to_couch.py Anexo [1](version anterior)

Para el primer fallo se corrigió el error creando bases para cada índice en mongoDB, en vista que existía conflictos si existían 2 o más colecciones en una base al momento de trasladarse a couchDB.

El segundo conflicto no pudo ser correguido a tiempo y por instancia final se desarrolló el script mongo2couch2.py partiendo de scripts funcionales en clases pasadas, en esta ocasión por motivos ya explicados se comentó la línea 56 dentro del código, omitiendo la validación de la id y $oid para guardar en couchDB.

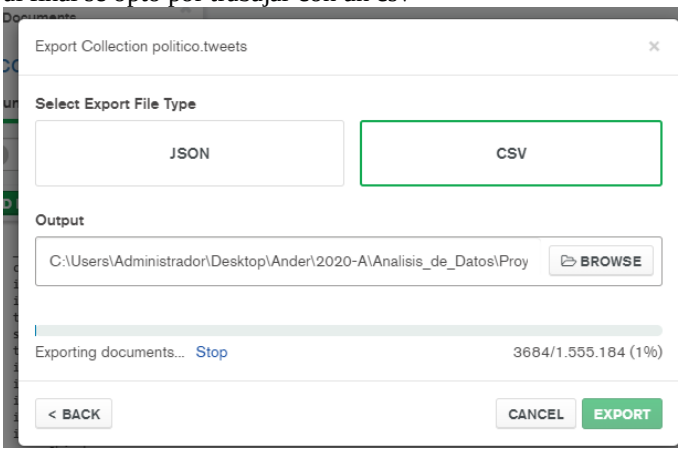
Al cambiarnos a este nuevo script se tuvo la limitante de pasar las bases 1 por 1, y como se cambió el diseño de las bases por el conflicto anterior, esto termino afectando a la arquitectura inicial, mas no el resultado final, pues, las bases fueron condensadas en elasticsearch.

```
44
45
46  for db in DBS:
47      if db not in ('admin', 'local','config'):
48          cols = CLIENT[db].list_collection_names()
49          for col in cols:
50              print('Querying documents from collection {} in database {}'
51              for x in CLIENT[db][col].find():
52                  try:
53
54                      documents=json.loads(json_util.dumps(x))
55
56                      #documents["_id"]=str(documents["_id"]["$oid"])
57
58
59                      print(documents)
60                      doc=dbc.save(documents)
61
62                  except TypeError as t:
63
64                      print('current document raised error: {}'.format(t))
65                      SKIPPED.append(x)  # creating list of skipped docume
66                      continue    # continue to next document
67                  except Exception as e:
68                      raise e
```

mapping.json Anexo [1]

Otro problema grave que presento este script fue el hecho de conflicto si por alguna razon se llegaba a interrumpir la ejecucion del codigo, pues al intentar correrlo denuevo existirai conflicto con los datos ya ingresados

Esto nos presentó un grave problema, sobre todo con la base con mayor cantidad de datos político en mongoDB, por lo que al final se optó por trabajar con un csv



mongodb_localhost

XIV. REFERENCIAS

[1] *G. PowerData, "Data lake: definición, conceptos clave y mejores prácticas", Powerdata.es, 2020. [Online]. Available: https://www.powerdata.es/data-lake#:~:text=Un%20data%20lake%20es%20un%20repositorio%20de%20almacenamiento%20que%20contienen,plana%20para%20almacenar%20los%20datos*

[2] D. Ortiz, "¿Qué es un dashboard y para qué se usa? (2020)", Cyberclick.es, 2020. [Online]. Available: https://www.cyberclick.es/numerical-blog/que-es-un-dashboard.

[3] "Home | Projectlibre", Projectlibre.com, 2020. [Online]. Available: https://www.projectlibre.com/

[4] *"Welcome to Python.org", Python.org, 2020. [Online]. Available: https://www.python.org/*

[5] *"Apache CouchDB", Couchdb.apache.org, 2020. [Online]. Available: https://couchdb.apache.org/.*

[6] "La base de datos líder del mercado para aplicaciones modernas", MongoDB, 2020. [Online]. Available: https://www.mongodb.com/es

[7] *"PostgreSQL: The world's most advanced open source database", Postgresql.org, 2020. [Online]. Available: https://www.postgresql.org/*

[8] "XAMPP" 2020. [Online]. Available: https://www.apachefriends.org/es/index.html.

[9] "Logstash: Recopila, parsea y transforma logs | Elastic", Elastic, 2020. [Online]. Available: https://www.elastic.co/es/logstash.

[10] "Elasticsearch: El motor de búsqueda y analítica distribuido oficial | Elastic", Elastic, 2020. [Online]. Available: https://www.elastic.co/es/elasticsearch/.

[11] "Kibana: Explora, visualiza y descubre datos | Elastic", Elastic, 2020. [Online]. Available: https://www.elastic.co/es/kibana

[12] "Tableau Public", Tableau Public, 2020. [Online]. Available: https://public.tableau.com/en-us/s/

[13] "Visualización de datos | Microsoft Power BI", Powerbi.microsoft.com, 2020. [Online]. Available: https://powerbi.microsoft.com/es-es/

[14] "VPN.net – Hamachi by LogMeIn", Vpn.net, 2020. [Online]. Available: https://www.vpn.net/.

[15] "Response Codes", Developer.twitter.com, 2020. [Online]. Available: https://developer.twitter.com/ja/docs/basics/response-codes.

XV. ANEXOS

[1] "stealth14 / data-lake", GitHub , 2020. [Onlibe]. Available: https://github.com/stealth14/data-lake.git.