EXTRACCIÓN Y OBTENCIÓN DE DATOS

IVONNE VIRGINIA VEGA PEDRAZA
Ivonne.vega01@epn.edu.ec

JOSE GEOVANNY DIAS AMAGUAÑA
Jose.dias@epn.edu.ec

ESCUELA POLITÉCNICA NACIONAL
ESCUELA DE FORMACIÓN DE TECNÓLOGOS

***RESUMEN*— En el presente documento se mostrará el diseño de un Data Warehouse en el cual convergen diferentes fuentes de información como Tableau, Twitter y Data.gov**

I. INTRODUCCIÓN

Un data warehouse es un repositorio unificado para todos los datos que recogen los diversos sistemas de una empresa. El repositorio puede ser físico o lógico y hace hincapié en la captura de datos de diversas fuentes sobre todo para fines analíticos y de acceso. [1]

Tableau es la plataforma de análisis integral más eficaz, segura y flexible para los datos. Además es la única plataforma de inteligencia de negocios que convierte los datos en información útil. [2]

Twitter es un servicio de comunicación bidireccional con el que puedes compartir información de diverso tipo de una forma rápida, sencilla y gratuita. [3]

Data.gov es un sitio web del gobierno de los EE.UU, apunta a mejorar el acceso público a conjuntos de datos de alto valor, legibles por máquina, generados por el Poder Ejecutivo del Gobierno Federal. El sitio es un depósito de información del gobierno federal, estatal, local y tribal, disponible para el público. [3]

II. OBJETIVOS

- Obtener información de tres fuentes distintas.
- Unificar la información de las diferentes fuentes.
- Visualizar en una interfaz web la información obtenida.

III. DESARROLLO

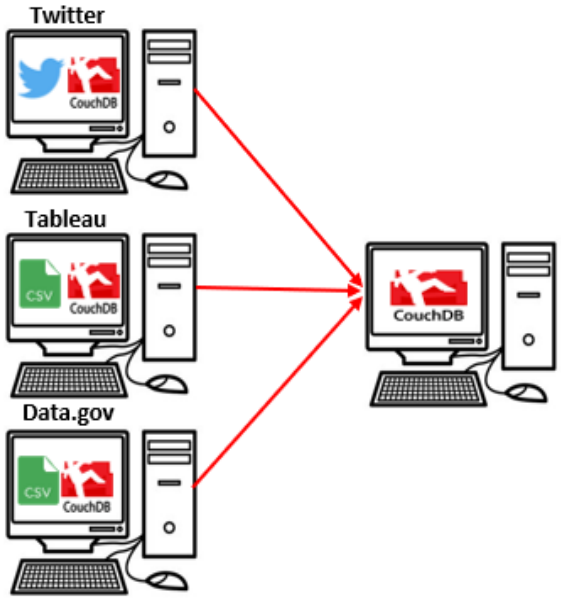
Se han seleccionado 3 temas para la recolección de datos:

1. Mascotas.
2. Entretenimiento.
3. Tecnología.

Para realizar este proyecto se utilizó los siguientes recursos.

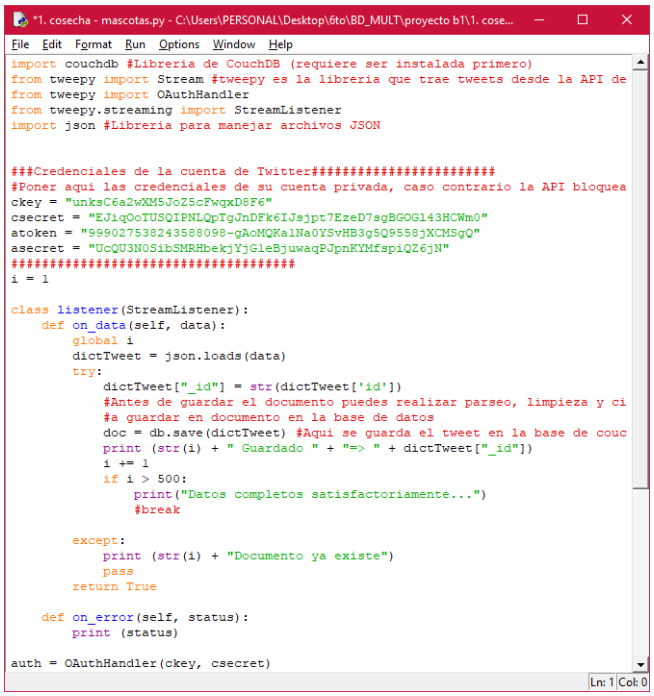
1. CouchDB 2.2.
2. Phython 5.0
3. USBWebserver v8.6 (Portable).
4. Editores de texto (Brackets).
5. Navegador Web

A) ARQUITECTURA



B) Recolección de información de Twitter.

Para la recolección de datos de Twitter se utilizó el siguiente script:



```
*1. cosecha - mascotas.py - C:\Users\PERSONAL\Desktop\6to\BD_MULT\proyecto b1\1. cose...
File Edit Format Run Options Window Help
import couchdb #Libreria de CouchDB (requiere ser instalada primero)
from tweepy import Stream #tweepy es la libreria que trae tweets desde la API de
from tweepy import OAuthHandler
from tweepy.streaming import StreamListener
import json #Libreria para manejar archivos JSON


###Credenciales de la cuenta de Twitter#########################
#Poner aqui las credenciales de su cuenta privada, caso contrario la API bloquea
ckey = "unksC6a2wXM5JoZ5cFwqxD8F6"
csecret = "EJiqOoTUSQIPNLQpTgJnDFk6IJsjpt7EzeD7sgBGOG143HCWm0"
atoken = "999027538243588098-gAoMQKalNaOYSvHB3g5Q9558jXCMSgQ"
asecret = "UcQU3N0SibSMRHbekjYjGleBjuwaqPJpnKYMfspiQZ6jN"
#####################################
i = 1

class listener(StreamListener):
    def on_data(self, data):
        global i
        dictTweet = json.loads(data)
        try:
            dictTweet["_id"] = str(dictTweet['id'])
            #Antes de guardar el documento puedes realizar parseo, limpieza y ci
            #a guardar en documento en la base de datos
            doc = db.save(dictTweet) #Aqui se guarda el tweet en la base de couc
            print (str(i) + " Guardado " + "=> " + dictTweet["_id"])
            i += 1
            if i > 500:
                print("Datos completos satisfactoriamente...")
                #break

        except:
            print (str(i) + "Documento ya existe")
            pass
        return True

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

auth = OAuthHandler(ckey, csecret)
```

En el cual tenemos que setear la URL del servidor de couchDB:

```
server = couchdb.Server('http://ivonne:ivonne@localhost:5984/')
```

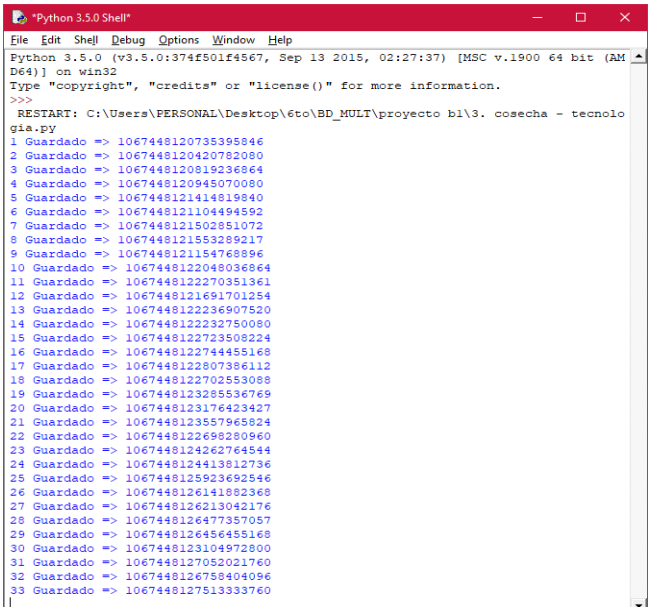
Crear la base de datos:

```
db = server.create('mascotas_1')
```

Definir el filtro por el cual se quiere buscar los tweets:

```
twitterStream.filter(track=["mimascota","perros","gatos","pets","dogs","cats"])
```

Al momento de ejecutar los scripts se pudo observar como se iba guardando los datos en couchDB.



```
*Python 3.5.0 Shell*
File Edit Shell Debug Options Window Help
Python 3.5.0 (v3.5.0:374f501f4567, Sep 13 2015, 02:27:37) [MSC v.1900 64 bit (AM
D64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
 RESTART: C:\Users\PERSONAL\Desktop\6to\BD_MULT\proyecto b1\3. cosecha - tecnolo
gia.py
1 Guardado => 1067448120735395846
2 Guardado => 1067448120420782080
3 Guardado => 1067448120819236864
4 Guardado => 1067448120945070080
5 Guardado => 1067448121414819840
6 Guardado => 1067448121104494592
7 Guardado => 1067448121502851072
8 Guardado => 1067448121553289217
9 Guardado => 1067448121154768896
10 Guardado => 1067448122048036864
11 Guardado => 1067448122270351361
12 Guardado => 1067448121691701254
13 Guardado => 1067448122236907520
14 Guardado => 1067448122232750080
15 Guardado => 1067448122723508224
16 Guardado => 1067448122744455168
17 Guardado => 1067448122807386112
18 Guardado => 1067448122702553088
19 Guardado => 1067448123285536769
20 Guardado => 1067448123176423427
21 Guardado => 1067448123557965824
22 Guardado => 1067448122698280960
23 Guardado => 1067448124262764544
24 Guardado => 1067448124413812736
25 Guardado => 1067448125923692546
26 Guardado => 1067448126141882368
27 Guardado => 1067448126213042176
28 Guardado => 1067448126477357057
29 Guardado => 1067448126456455168
30 Guardado => 1067448123104972800
31 Guardado => 1067448127052021760
32 Guardado => 1067448126758404096
33 Guardado => 1067448127513333760
```

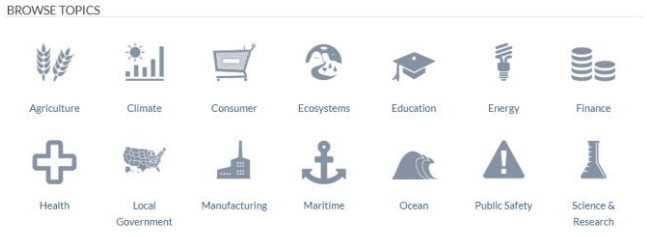
entretenimiento_1	1.5 MB	569
mascotas_1	1.5 MB	547
tecnologia_1	1.6 MB	571

C) Recolección de datos de Tableau.

Para obtener los datos de Tableau se ingreso a la siguiente página: https://www.tableau.com/es-es , en la cual se procedió a descargar los archivos Excel.

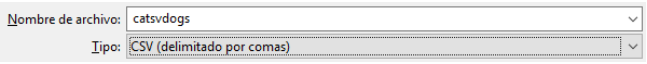
D) Recolección de datos en Data.gov

Para obtener los datos de Data.gov se ingresó a la siguiente página:
https://www.data.gov/ en la cual se puede escoger datos entre diversos temas, dichos documentos pueden ser archivos CSV o Excel.

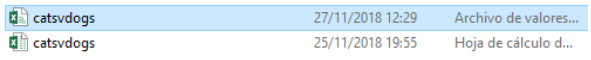


E) Transformar archivos Excel a CSV y guardar en CouchDB con formato JSON.

Primero tenemos que transformar los archivos de Excel en CSV. Simplemente al archivo *.xlsx lo guardamos como CSV (delimitados por comas).

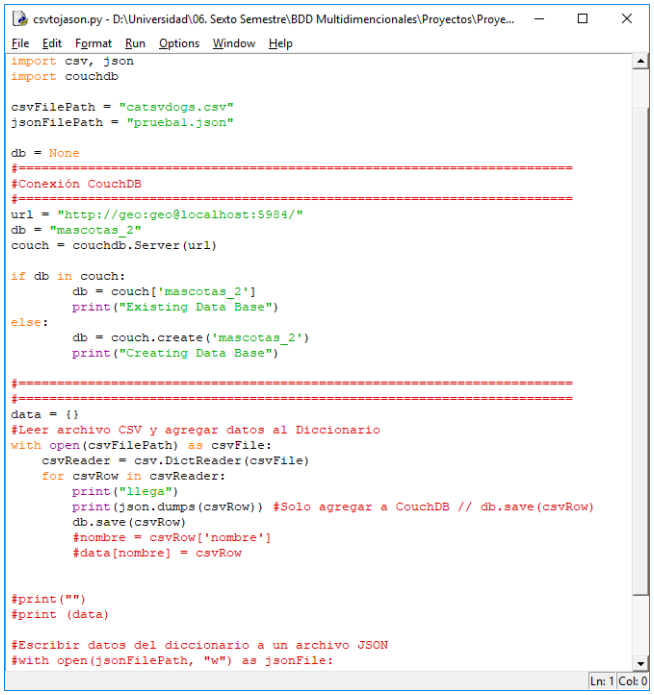


Se nos guarda el archivo en *.csv con el cual podremos ingresar a nuestra base de datos NoSQL.



F) Ingreso de datos de archivo CSV a CouchDB.

Para realizar este proceso hemos escrito un pequeño Script en lenguaje Python.



En el cual importamos las librerías necesarias para poder ejecutar el código y no nos marque error.

```
import csv, json
import couchdb
```

Se ingresa el nombre del archivo con la extensión *.csv, el cual contiene el dataset que se guardará en la base de datos.

```
csvFilePath = "catsvdogs.csv"
```

Realizamos la conexión a CouchDB y creamos la base de datos con el nombre seleccionado.

```
url = "http://geo:geo@localhost:5984/"
db = "mascotas_2"
couch = couchdb.Server(url)

if db in couch:
        db = couch['mascotas_2']
        print("Existing Data Base")
else:
        db = couch.create('mascotas_2')
        print("Creating Data Base")
```

El Script leerá cada línea del documento CSV en formato JSON e irá guardando automáticamente en CouchDB mediante un ciclo “For”.

```
#Leer archivo CSV y agregar datos al Diccionario
with open(csvFilePath) as csvFile:
    csvReader = csv.DictReader(csvFile)
    for csvRow in csvReader:
        print("llega")
        print(json.dumps(csvRow)) #Solo agregar a CouchDB // db.save(csvRow)
        db.save(csvRow)
```

Ahora podemos observar cómo se crea la base de datos y como se van guardando los datos:

entretenimiento_1	1.5 MB	569
entretenimiento_2	175.6 KB	235
entretenimiento_3	68.5 KB	194
mascotas_1	1.5 MB	547
mascotas_2	24.4 KB	50
mascotas_3	304.2 KB	281
tecnologia_1	1.6 MB	571
tecnologia_2	491.9 KB	1898
tecnologia_3	2.2 MB	5899

Fuente: Tableau

Tema: Mascotas

Tema: Entretenimiento

Fuente: Data.gov

Tema: Mascotas

Tema: Tecnología

Tema: Entretenimiento

Tema: Tecnología

Cada vez que ejecutábamos un Script se visualizaba como los datos se leían en formato JSON y se guardaban en la base de datos NoSQL de CouchDB.

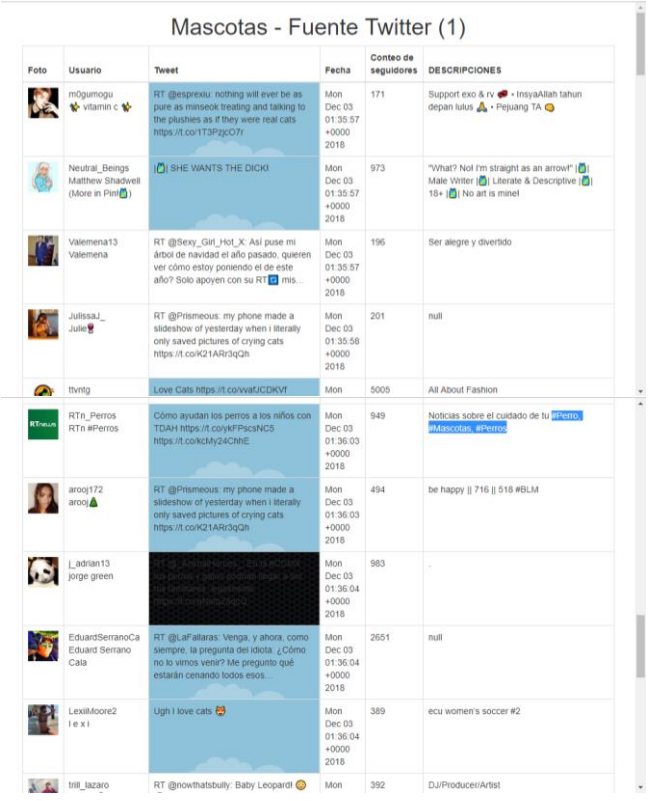
G) APLICACIÓN WEB.

Se creó una interface interactiva para poder visualizar la información recopilada de las diferentes fuentes.

La interface cuanta con un menú de navegación que permitirá al usuario navegar de forma fácil y segura.

Se puede seleccionar el tema de interés y mediante un selector de etiquetas elegir la fuente de donde fue obtenida la información.

Seleccionamos la fuente y la interfaz nos direccionara a una nueva página donde mostrara la información de guardada en nuestra base de datos NoSQL de CouchDB.



IV. CONCLUSIONES

Para diseñar una buena arquitectura de un Data Warehouse es necesario conocer bien los requerimientos del negocio y hacer un estudio profundo de las fuentes externas que nos van a suministrar los datos.

Tener un Data Warehouse dentro de una organización es muy útil debido a que ayuda y facilita la toma de decisiones ya que permite tener un buen control sobre toda la información que necesita la organización para poder tomar decisiones, basándose en datos confiables y que pueden tener a la mano en cualquier momento.

V. REFERENCIAS

[1] «PowerData,» PowerData, [En línea]. Available: https://www.powerdata.es/data-warehouse. [Último acceso: 29 11 2018].

[2] «Tableau,» Tableau, [En línea]. Available: //www.tableau.com/es-es/products/what-is-tableau#UWd33zZDTkMuUIYY.99. [Último acceso: 29 11 2018].

[3] «WebEmpresa,» [En línea]. Available: https://www.webempresa.com/blog/que-es-twitter-como-funciona.html. [Último acceso: 29 11 2018].

[4] «Wikipedia,» [En línea]. Available: https://en.wikipedia.org/wiki/Data.gov. [Último acceso: 29 11 2018].

[5] M. Rouse, «SearchDataCenter,» [En línea]. Available: https://searchdatacenter.techtarget.com/es/definicion/CouchDB. [Último acceso: 03 11 2018].

VI. ANEXO

https://github.com/GeovannyDias/Datawarehouse-with-CouchDB