
PROYECTO 3

202001574 – Josue Daniel Solis Osorio

Resumen

El siguiente proyecto se realizó con el fin de poder realizar una Interfaz de programación de aplicaciones a través de Django como framework, frontend y Flask como backend

Este se utilizó para realizar una Interfaz de programación de aplicaciones con la cual permitiese leer archivos.xml y con esto obtener una serie de datos que se solicitan a través del mismo por medio de la librería xml.dom.minidom la cual fue fundamental a la hora de realizar este proyecto; se necesitó una librería para poder leer los xml, también se utilizó la librería regex de Python para poder identificar las expresiones regulares de el archivo xml

Se utilizo Python (versión 3.10.2 de 64-bits) con el de poder desarrollar la Interfaz de programación de aplicaciones ya que se consumió utilizando el protocolo HTTP.

Abstract

The following project was carried out in order to be able to make an application programming interface through Django as a framework, frontend and Flask as a backend.

This was obtained to make an Application Programming Interface with which it would allow to read .xml files and with this obtain a series of data that is requested through it through the xml.dom.minidom library which was fundamental to the time to make this project; a library was needed to be able to read the xml, the Python regex library was also obtained to be able to identify the regular expressions of the xml file

Python (64-bit version 3.10.2) was used to be able to develop the application programming interface since it was consumed using the HTTP protocol.

Palabras clave

API lectura xml

Leer xml minidom

API django-flask

Keywords

xml reading Read

Read xml minidom

django-flask API

Introducción

Este Proyecto se realizo con el fin de poder ampliar los conocimientos sobre la programación utilizando a través de lenguaje Python que pueda ser consumida utilizando el protocolo HTTP, Django utilizando sus funciones como framework y Flask como backend

Por medio de un encargo como ejemplo ya que con esto se obtuvo conocimiento acerca de comoutilizar archivos xml extrayendo lo mas esencial de lomismo para tener una correcta lectura a la hora de realizar una petición o una modificación de estas y posteriormente con su correcta ejecución

La información se manipulo a través de las listas que se encuentran en el lenguaje de progrmacion Python ya que esta es una estructura de datos que consiste en un conjunto de nodos enlazados secuencialmente y facilita la el tratar con la información de los archivos xml de una manera mas

Desarrollo del tema

La empresa Tecnologías Chapinas, S.A. está desarrollando una herramienta que sea capaz de analizar contenido de redes sociales y establecer el sentimiento de los usuarios respecto a una empresa y los servicios que provee.

Para lograr este fin, la empresa Tecnologías Chapinas, S.A. lee mensajes en redes sociales y construye estructuras de texto con el siguiente formado:

Lugar y Fecha: Guatemala, dd/mm/yyyy hh24:mi
Usuario: nombre_usuario_sin_espacios
Red social: nombre_red_social_sin_espacios
Mensaje escrito por el usuario en una red social

Figura 1. instruccion de entrada
Fuente: Enunciado-Proyecto3

La empresa Tecnologías Chapinas, S.A. ha creado una estrategia para establecer si un mensaje tiene un sentimiento positivo, negativo o neutro a través de la creación de un diccionario de datos que determine palabras que puedan calificar un mensaje como positivo o negativo, en caso de no tener palabras del diccionario de datos específico, o bien, que la cantidad de palabras con sentimientos positivos y negativos sean iguales, entonces, se considera que el mensaje es neutro. En este mismo diccionario de datos, es posible determinar los nombres de las empresas y sus servicios que se están analizando para determinar si en un momento dado, las redes sociales están mostrando un sentimiento positivo o negativo del mismo.

Para iniciar el programa primero se debe de introducir un archivo xml



Figura 3. Ejemplo archivo xml
Fuente: Enunciado-Proyecto3

Una vez obtenido el archivo xml se procede a colocar en la bandeja de entrada de la API



Figura 3. Entrada API
Fuente: elaboración propia 2022

Una vez se tenga el archivo xml en la bandeja de entrada se procede a seleccionar el botón arriba a la derecha llamado “cargar archivo” esto lo que hace es realizar una petición http de django que se utiliza como frontend a Flask que se utiliza como backend

Realizado el paso anterior se procederá a mostrar los datos obtenidos del xml ingresado en cuestión en el textarea de abajo llamado ”Salida” en este text área se mostrar datos como: nombre de la empresa, mensajes ya sean positivos, negativos o neutrales cuantas veces se detecto el alias de un servicio en los texto enviados desde el xml

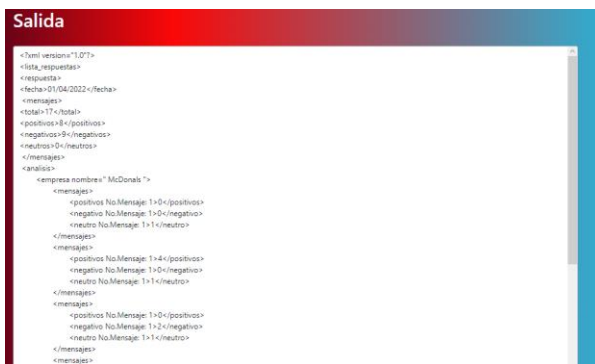


Figura 4. Salida API
Fuente: elaboración propia
2022

Los botones de la parte del header tienen las siguientes funciones:

Cargar Archivo: Se desplegará una pantalla para gestionar la carga de los archivos de entrada con extensión .xml con una o varias solicitudes de clasificación. Se especifica en la sección de archivos de entrada y salida.

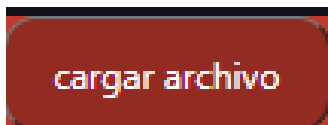


Figura 5. Boton cargar
archivo Fuente: elaboración
propia 2022

Peticiones: En este apartado se debe de tener las siguientes opciones:

Consultar Datos: Al seleccionar esta opción se deben de consultar los datos almacenados en el último archivo de respuesta y se mostrarán los datos en un recuadro de texto de salida.

Resumen de clasificación por fecha: Al seleccionar esta opción se podrá elegir la fecha por la cual se requiere filtrar y si se desea incluir una empresa o todas las empresas, entonces se debe de mostrar gráficamente un resumen de mensajes clasificados. Debe presentar el total de mensajes, el total de mensajes positivos, el total de mensajes negativos y el total de mensajes neutros para la fecha y empresa seleccionados.

Resumen por rango de fechas: Al seleccionar esta opción se podrá elegir un rango de fechas por la cual se requiere filtrar, luego podrá elegir si desea ver una empresa o todas las empresas. Se deberá presentar gráficamente por cada fecha y empresa el total de mensajes, el total de mensajes positivos, el total de mensajes negativos y el total de mensajes neutros.

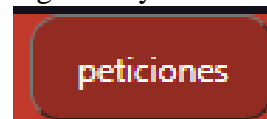


Figura 6. Boton Peticiones
Fuente: elaboración propia
2022

Ayuda: desplegará 2 opciones, una para visualizar información del estudiante y otra para visualizar la documentación del programa.

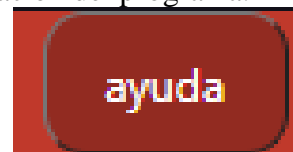


Figura 7. Boton Ayuda
Fuente: elaboración propia
2022

