Universidad de San Carlos de Guatemala
Escuela de Ingeniería en Ciencias y Sistemas, Facultad de Ingeniería
Introducción a la programación y computación 2, 1er. Semestre 2022.

# DESARROLLO DE APLICACIONES WEB CON DJANGO Y FLASK

**202006688 – Karla Ernestina González Polanco**

## Resumen

El tercer proyecto del curso “Introducción a la Programación y Computación 2”, se basa en el uso de Django y Flask para la creación de una aplicación web capaz de recibir un archivo de entrada, el cual contendrá un diccionario y una lista de mensajes, los cuales tendrán que ser clasificados según los parámetros especificados en el diccionario. También existe la posibilidad de analizar mensajes individuales y con la información obtenida con las peticiones anteriores, generar reportes respecto a una fecha o rango de fechas, conteniendo su gráfica o información en un archivo PDF. Para esta ocasión se utilizó Flask como framework para definir las acciones de los endpoints, y el uso de Django fue exclusivamente para el desarrollo de los aspectos visuales de la aplicación y las acciones del envío de información desde el frontend, que es donde el usuario tendrá acceso a la información, y el backend, donde se hará todo el proceso de análisis y manejo de la información que se ingrese en el sistema.

## Palabras clave

*API: Interfaz de programación de aplicaciones.*
*DJANGO/FLASK: Framework para desarrollo web.*
*FRONTEND: Diseño visual y envío de la data.*
*BACKEND: Parte de la app que maneja la data.*

## *Abstract*

*The third project of the course " Introducción a la Programación y Computación 2", is based on the use of Django and Flask for the creation of a web application capable of receiving an input file, which will contain a dictionary and a list of messages, which will have to be classified according to the parameters specified in the dictionary. There is also the possibility of analyzing individual messages and with the information obtained with the previous requests, generating reports regarding a date or range of dates, containing its graph or information in a PDF file. For this occasion, Flask was used as a framework to define the actions of the endpoints, and the use of Django was exclusively for the development of the visual aspects of the application and the actions of sending information from the frontend, which is where the user will have access to information, and the backend, where the entire process of analysis and management of the information entered into the system will be carried out.*

## *Keywords*

*API: Application Programming Interfaces*
*DJANGO/FLASK:Web development framework.*
*FRONTEND:Visual design, data sending..*
*BACKEND: Data management and analysis.*

## Introducción

Para el desarrollo del proyecto fue necesario el uso de los framework Flask y Django. En este caso, Flask se utilizó como backend, con él y con el desarrollo de funciones fue posible la creación de los endpoints necesarios para que las peticiones posibles a los datos, pudieran realizarse de manera correcta haciendo uso de las funciones anteriormente mencionadas, las cuales tomaban la información que se tenía en la data y las manejaba para retornar algún tipo de archivo o respuesta para que pudiera reflejarse del lado del cliente con Django, en donde fueron establecidas las diferentes views posibles dependiendo de la acción que el cliente desee realizar en la aplicación, entre las cuales están las de cargar, las de peticiones, ayuda, etc. En Django fue necesario crear las funciones que llamarían a cierto endpoint del backend para entonces poder obtener una respuesta referente al tipo de petición que se requiera.

## 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. 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 realizar este análisis deberá cargarse al sistema un archivo que contenga una estructura como la siguiente:

```
<?xml version="1.0"?>
<solicitud_clasificacion>
  <diccionario>
    <sentimientos_positivos>
      <palabra> bueno </palabra>
          <palabra> excelente </palabra>
          <palabra> cool </palabra>
          <palabra> satisfecho </palabra>
          ...
    </sentimientos_positivos>
    <sentimientos_negativos>
      <palabra> malo </palabra>
          <palabra> pésimo </palabra>
          <palabra> triste </palabra>
          <palabra> molesto </palabra>
          <palabra> decepcionado </palabra>
          <palabra> enojo </palabra>
          ...
    </sentimientos_negativos>
    <empresas_analizar>
      <empresa>
        <nombre> USAC </nombre>
        <servicio nombre="inscripción">
                <alias> inscribí </alias>
                <alias> inscrito </alias>
              </servicio>
              <servicio nombre="asignación">
                <alias> asignado </alias>
        </servicio>
        <servicio nombre="graduación"> </servicio>
        ...
         </empresa>
         ...
    </empresas_analizar>
  </diccionario>
  <lista_mensajes>
    <mensaje>
      Lugar y fecha: Guatemala, 01/04/2022 15:01 Usuario:
         map0001@usac.edu   Red social: Twitter
         El servicio en la USAC para inscripción fue muy bueno y me siento muy satisfecho.
    </mensaje>
      <mensaje>
      Lugar y fecha: Guatemala,
          01/04/2022 15:20 Usuario: map0002@usac.edu   Red social: Facebook Hoy me inscribí
          en la USAC, no encontré parqueo e inicié molesto mi gestión, luego tuve que hacer
          cola y me indicaron que era la cola incorrecta, esto me enojó mucho y mejor me
          fui.
    </mensaje>
```

Para realizar el análisis de los mensajes fue necesario crear en el lado del backend diferentes endpoints con funciones cuyo resultado sea equivalente a lo que el usuario está solicitando del lado del frontend.

- Backend:

| Ruta | Método | Return |
|---|---|---|
| /ConsultarDatos | POST | XML |

El primer endpoint implementado lleva la función:

- add_solicitud: en ella se hace la descomposición del archivo de entrada según las etiquetas que este debería tener. Esta función se aboca de otras definidas en el archivo de manager. Entre ellas:
    - agregarSPositivos: agrega a la lista de sentimientos positivos todas aquellas palabras que presente el diccionario.
    - agregarSNegativos: agrega a la lista e sentimientos negativos todas aquellas palabras que presente el diccionario.
    - agregarEmpresa: agrega a la lista de empresas todas aquellas que se presenten en el apartado de empresas a analizar del diccionario.
    - agregarServicios: va asignando a cada empresa los servicios que esta tendrá disponibles y seguidamente para cada servicio se agregarán los alias en su lista.

  Luego de agregar toda la información del diccionario a las listas correspondientes. Se analizan los mensajes que vengan en el archivo de entrada. Para ello se usan los métodos:

    - getTSentimiento: toma el mensaje y cuenta las palabras que hagan match con alguna de la lista de sentimientos positivos o negativos. Suma en 1 a una variable dependiendo del tipo de palabra con la que haga match, y al final con unas comparaciones entre variables se decide si el mensaje es neutro, positivo o negativo.
    - getFecha: hace uso de expresiones regulares para encontrar la fecha en la que se envió el mensaje.
    - getEmpresa: retorna la empresa con la que hizo match el mensaje que se analizó.
    - getServicio: buscar y hace match cuando encuentre el servicio de la empresa a la que se refiere el mensaje, haciendo uso de la búsqueda de algún alias o en sí en nombre del servicio.

Cuando se tiene toda la información anterior de un mensaje a analizar, se envía toda esa información y con el método writeXML que recibe la lista de fechas del documento y se escribe un archivo XML que tendrá una estructura como la siguiente:

```xml
<?xml version="1.0" ?>
<lista_respuestas>
    <respuesta>
        <fecha>01/04/2021</fecha>
        <mensajes>
            <total>2</total>
            <positivos>1</positivos>
            <negativos>1</negativos>
            <neutros>0</neutros>
        </mensajes>
        <analisis>
            <empresa nombre="USAC">
                <mensajes>
                    <total>1</total>
                    <positivos>1</positivos>
                    <negativos>0</negativos>
                    <neutros>0</neutros>
                </mensajes>
                <servicios>
                    <servicio nombre="inscripción">
                        <mensajes>
                            <total>1</total>
                            <positivos>1</positivos>
                            <negativos>0</negativos>
                            <neutros>0</neutros>
                        </mensajes>
                    </servicio>
                </servicios>
            </empresa>
            <empresa nombre="UVG">
                <mensajes>
                    <total>1</total>
```

| Ruta | Método | Return |
|---|---|---|
| /ProcesarMensaje | GET | XML |

En este se hace uso únicamente de la función analizarMensaje, la cual hace uso de expresiones regulares para encontrar la fecha del mensaje y ya que la estructura inicial del mensaje tiene la misma forma, se hace uso de expresiones regulares que separarán la entrada por signos ":", y la posición 3 de la lista corresponderá al nombre de la red social de la que proviene el mensaje. A continuación, se usan expresiones regulares para determinar si viene un correo, en caso de que no, se toma la posición 2 de la cadena que pasó por la expresión regular de ":". Se busca la empresa y servicio y se crea el archivo XML con las especificaciones del mensaje.

| Ruta | Método | Return |
|---|---|---|
| /ConsultaFecha | GET | JSON |

Recibe un archivo json del cual, en la primera posición se espera la fecha a analizar y en la segunda las empresas que se incluirán. Con los datos que se tienen, se va creando un archivo json con el nombre de la empresa, el total, número de mensajes positivos negativos y neutros.

Se retorna el archivo JSON y se envía al frontend para hacer una gráfica con los datos.

| Ruta | Método | Return |
|---|---|---|
| /ConsultaRangoFecha | GET | JSON |

Recibe un archivo json del cual, en la primera posición se espera la fecha inicial del rango a analizar, en la segunda la fecha inicial del rango y en la última, las empresas que se incluirán. Con los datos que se tienen, se va creando un archivo json con el nombre de la empresa, el total, número de mensajes positivos negativos y neutros.

Se retorna el archivo JSON y se envía al frontend para hacer una gráfica con los datos.

## Conclusiones

El desarrollo de aplicaciones web es un trabajo que requiere la capacidad de manejo de diversos frameworks que se adapten de mejor manera a la función que van a realizar para la api.

Al momento de crear una aplicación web es necesario definir inicialmente cuáles son los endpoints que se consumirán, ya que estos definirán las funciones lógicas de las que se valerá el problema para hacer el manejo y análisis de la información. Por otro lado, django es una opción muy interesante para el desarrollo visual que es con el que el usuario tendrá interacción.

El manejo de la información en archivos con extensión XML permite el aprendizaje de su manejo con diferentes tipos de lenguajes capaces de construir expresiones que puedan recorrerlo y de él extraer información importante para su posterior manejo en una aplicación.

Todos los conocimientos adquiridos con la realización del proyecto son piezas importantes para permitir el desarrollo de la lógica de implementación de nuevas estructuras y de hacer su correcto manejo para obtener los resultados necesarios al ejecutar su aplicación.

## Referencias bibliográficas

Phyton TM, (2001-2022). *Minimal DOM implementation.* Python Software Foundation.

Django Software Foundation, (2015). *Django documentation.* Django.

Flask, (2010). *Flask web development, one drop at a time).* Pallets.

## Anexos

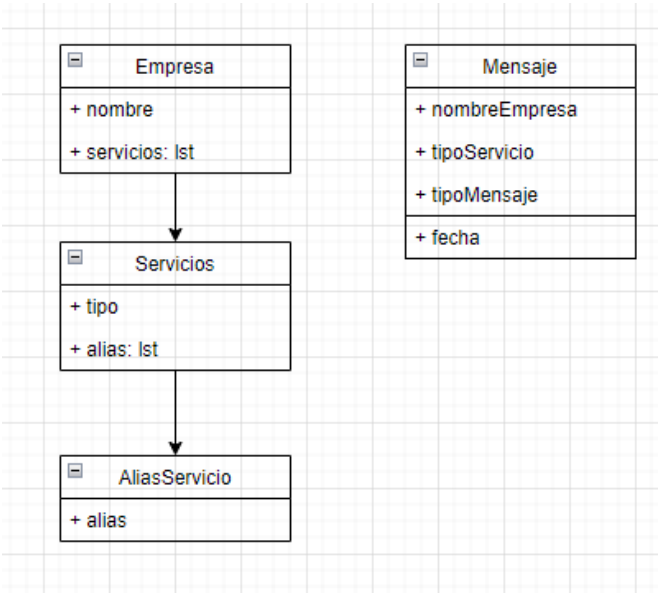


*Figura 4.* Diagrama UML.

Fuente: elaboración propia.