
Proyecto 3 – Tecnologías chapinas, S.A.

202247844 – Josué Samuel de la Cruz Medina

Resumen

El proyecto está centrado en la elaboración de una página web para la empresa Tecnologías Chapinas, S.A. La cual se encuentra diseñada para analizar mensajes en redes sociales y clasificar el sentimiento hacia los servicios que proporcionan diferentes empresas.

En este proyecto se carga un archivo XML desde el frontend, el cual contiene los diferentes aspectos que nos van a servir, entre ellos se encuentran: palabras positivas, palabras negativas, empresas, servicios y mensajes que fueron proporcionados por los distintos clientes.

Gracias a esto las empresas pueden mejorar sus servicios en función de la retroalimentación de los clientes en redes sociales.

En conclusión, gracias al proyecto las empresas tienen acceso a una herramienta técnica que tiene el fin de mejorar la reputación de la o las empresas, mejorando el impacto social de las mismas.

Palabras clave

Archivo XML, página web, análisis, sentimiento, diccionario.

Abstract

The project is focused on the development of a website for the company Tecnologías Chapinas, S.A. Which is designed to analyze messages on social networks and classify the sentiment towards the services provided by different companies.

In this project, an XML file is loaded from the frontend, which contains the different aspects that will be useful to us, among them are: positive words, negative words, companies, services and messages that were provided by the different clients.

Thanks to this, companies can improve their services based on customer feedback on social networks.

In conclusion, thanks to the project, companies have access to a technical tool that aims to improve the reputation of the company or companies, improving their social impact.

Keywords

XML file, web page, analysis, sentiment, dictionary.

Introducción

Actualmente las redes sociales son de vital importancia para todo tipo de empresa que desea seguir existiendo e incluso expandirse, ya que si logran tener un impacto positivo en ellas, obtendrán más clientes, por otro lado si el impacto es negativo podría afectarlas de tal manera que incluso quebrarían. Es importante que las empresas se auto perciban en redes sociales.

Para ayudar a las empresas se desarrolló un programa en Python, una página web. Con el uso de la herramienta Django para el Frontend y Flask para el backend. En esta página web se lee y procesa un archivo XML. Luego de esto se realiza otro XML de salida.

Se usaron estructuras de datos nativas en Python, junto a varias librerías que ayudaron en diferentes áreas.

Desarrollo del tema

Dicho anteriormente, este proyecto fue desarrollado en el lenguaje de programación Python, principalmente con la ayuda de dos frameworks del mismo: Django y Flask.

Al ser obligatorio el uso de Python, se pueden encontrar distintos videos en la red o material de apoyo para el uso de los frameworks.

Django fue utilizado para el frontend, donde por medio de templates y de un archivo llamado views.py, se logró diseñar y personalizar la página web

Por otro lado, Flask es un framework que fue utilizado para el backend. Flask ofrece una gran compatibilidad con Python y no resulta un mayor problema.

A. Django



Figura 1 Logo django

Fuente: <https://acesse.one/Y5KF0>

Django es un framework relativamente fácil de utilizar, si ya se tiene experiencia haciendo páginas web en Python.

Básicamente se crea la carpeta Templates donde se crean distintas plantillas en HTML que se van a usar con cada dirección declarada con anterioridad en un archivo views.py y luego en otro archivo llamado urls.py,

Se puede incrustar código en los mismos archivos HTML para la lógica del programa, de esta manera son más dinámicos.

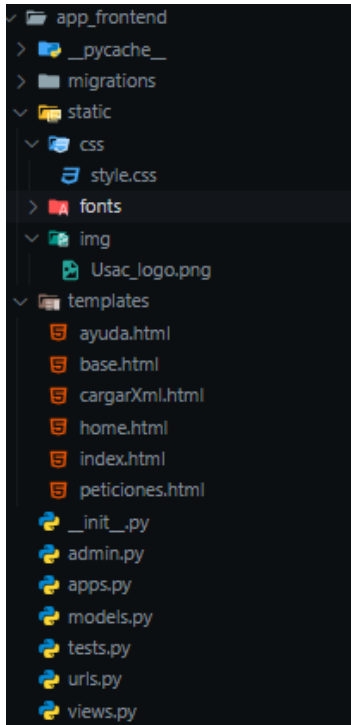


Figura 2 Archivos del frontend
Fuente: Elaboración propia

B. CSS

Para el diseño y personalización de las distintas etiquetas en HTML se utilizó un archivo CSS general para todos los archivos. De esta manera todos el decorado sería el mismo en las diferentes vistas que se tienen.

C. Flask



Figura 3 Logo Flask
Fuente: <https://acesse.one/IDOZt>

Dicho anteriormente, Flask se utilizó en el Backend. Para crear una dirección en Flask es tan sencillo como declarar la dirección junto a sus métodos, ya sea get o post y declarar una función debajo de esta misma que contenga la lógica o parte de la lógica de lo que se va a realizar en la página web, esto generalmente se realiza en el main de nuestro programa.

Aunque, al ser backend, Flask únicamente sirvió para las peticiones. Se colocaban las direcciones en el Frontend y desde ahí se mandaba la información para el backend para manipular los datos cargados.

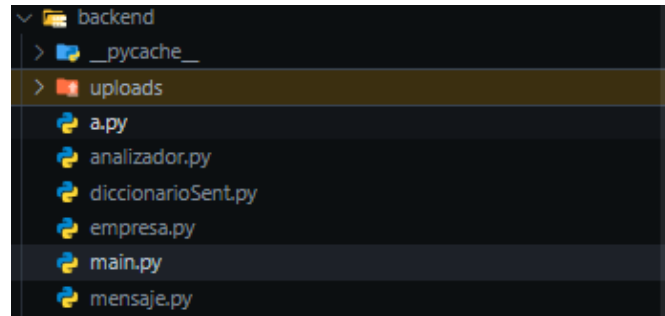


Figura 4 Clases del backend
Fuente: Elaboración Propia

D. Lógica

Al tener que utilizar Django y Flask, la conexión entre ambos fue un poco complicada al inicio, sin embargo se comprendió que era tan sencillo como mandar los request desde el Frontend a las diferentes direcciones creadas en el Backend.

Ya que fueron permitidas las estructuras de datos nativas, no fue necesario implementar unas propias. Principalmente fueron empleados las listas y diccionarios para guardar los datos.

Se utilizó Programación Orientada a Objetos (POO) para el guardado de las diferentes etiquetas que se encontraban en el archivo de entrada.

E. Librerías

En el proyecto fueron utilizadas distintas librerías, entre ellas ElementTree para el manejo de los archivos XML; reportlab para la generación del archivo PDF en la sección de peticiones.

Otra librería importante fue re, la cual sirvió para la creación y manejo de expresiones regulares o regex. Una librería digna de mención es la librería OS que sirvió para abrir y escribir sobre archivos.

F. Home

Al ejecutar el Frontend se nos brinda una dirección, la cual nos lleva a la página de inicio. Aquí podemos encontrar distintas opciones a elegir.

G. Instrucciones

Tenemos una vista con las instrucciones para el uso adecuado de la página web.

H. Peticiones

Luego de esto nos topamos con la vista Peticiones en el navBar, aquí puedes consultar datos, clasificar por fecha o rango de fechas, generar un archivo PDF de salida y mandar una prueba de mensaje.

I. Ayuda

En esta vista se puede ver la información del estudiante que programo la página web y donde se puede ver la documentación del programa.

J. Cargar archivo

Esta vista es la más importante, ya que desde esta podemos subir el archivo de entrada. Una vez subido, si es aceptado por el programa, redirige hacia Home. Si no es aceptado indica el formato que debe tener el archivo.

K. Procesamiento del archivo

Para procesar el archivo, se va leyendo y analizando. Si se encuentra con ciertas palabras reservadas el programa tiene indicado que debe realizar. El programa busca las diferentes palabras con diferentes sentimientos según el diccionario que se tenga.

Cuando llega a la parte de los mensajes, se utilizan tres expresiones regulares. La primera es para encontrar la fecha, ya que tenemos el formato DD/MM/AAAA, se configuró una expresión regular que detecta esto y guarda la fecha.

La segunda expresión regular es para encontrar al usuario dentro del mensaje. Se analiza si este tiene un símbolo @ en su usuario o caso contrario no lo tiene. Se detecta y se guarda.

La tercera expresión regular sirve para encontrar y guardar el nombre de la red social desde donde se escribió el mensaje. Una vez detectado el nombre se guarda.

Finalmente, se lee todo el mensaje y si se encuentra alguna palabra con sentimiento encontrada anteriormente, se le sumara la cantidad de 1 al tipo de sentimiento que tiene. Todas las palabras pasan por una función para quitar tildes y poner en minúscula las palabras.

Al finalizar la lectura del archivo de carga, se procede a generar un archivo de salida con el mismo formato XML. Este archivo lleva las distintas fechas que se encontraron. Con la cantidad de mensajes y sentimientos por cada uno. Al igual que con las diferentes empresas y sentimientos que tuvo cada una.

A partir de aquí, se puede ir a Peticiones, donde podremos pedir que se muestren mensajes dependiendo de la fecha o rango de fechas. También se puede generar un PDF con los datos del archivo de salida.

Conclusiones

En conclusión, realizar un proyecto con aspectos parecidos funciona de tal manera que las empresas cliente de este mismo comprendan su reputación y puedan mejorar sus estrategias, ya sea de marketing, atención al cliente, entre otros.

Durante la clasificación de los mensajes existe un rango de error todavía no determinado, se debe de someter a pruebas de carga el proyecto.

Teniendo en cuenta los distintos errores que surgieron a lo largo de la programación del proyecto, se

concluye que el producto se pudo haber mejorado en varios aspectos, pero logra tener éxito.

Referencias bibliográficas

C. J. Date, (1991). An introduction to Database Systems. Addison-Wesley Publishing Company, Inc.

Maldeadora (2018) Qué es frontend y backend: Características, Diferencias y Ejemplos, Platzi. Available at: <https://platzi.com/blog/que-es-frontend-y-backend/>

W3schools.com (no date) W3Schools Online Web Tutorials. Available at: <https://www.w3schools.com/0020>.

Apéndices:

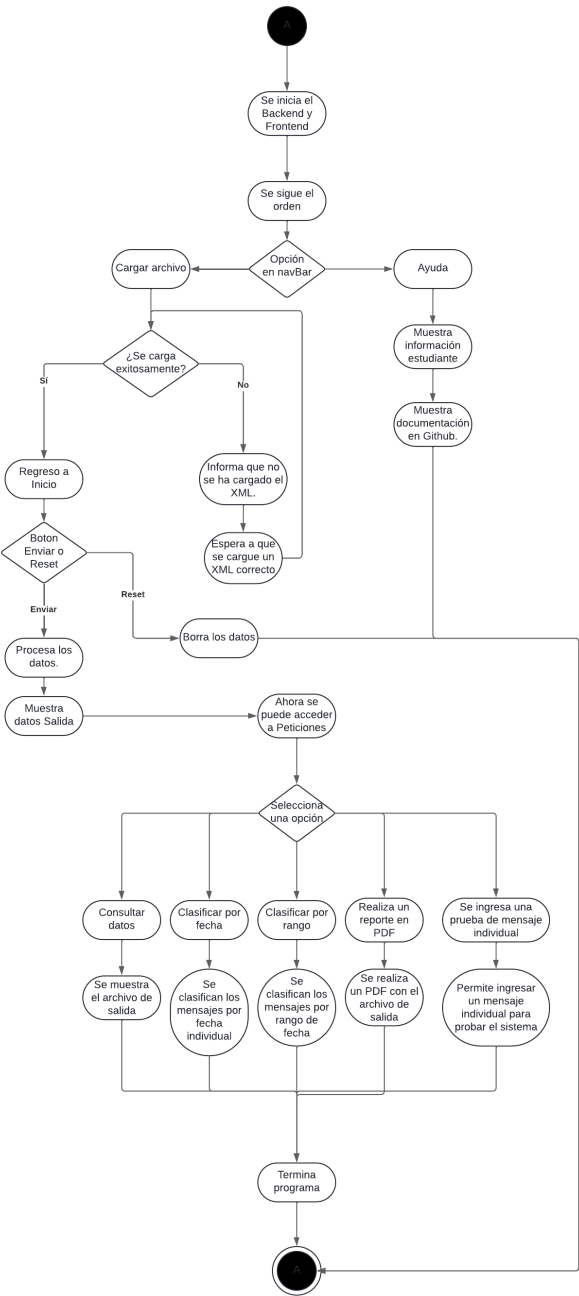


Figura 5 Diagrama de actividades
Fuente: Elaboración propia

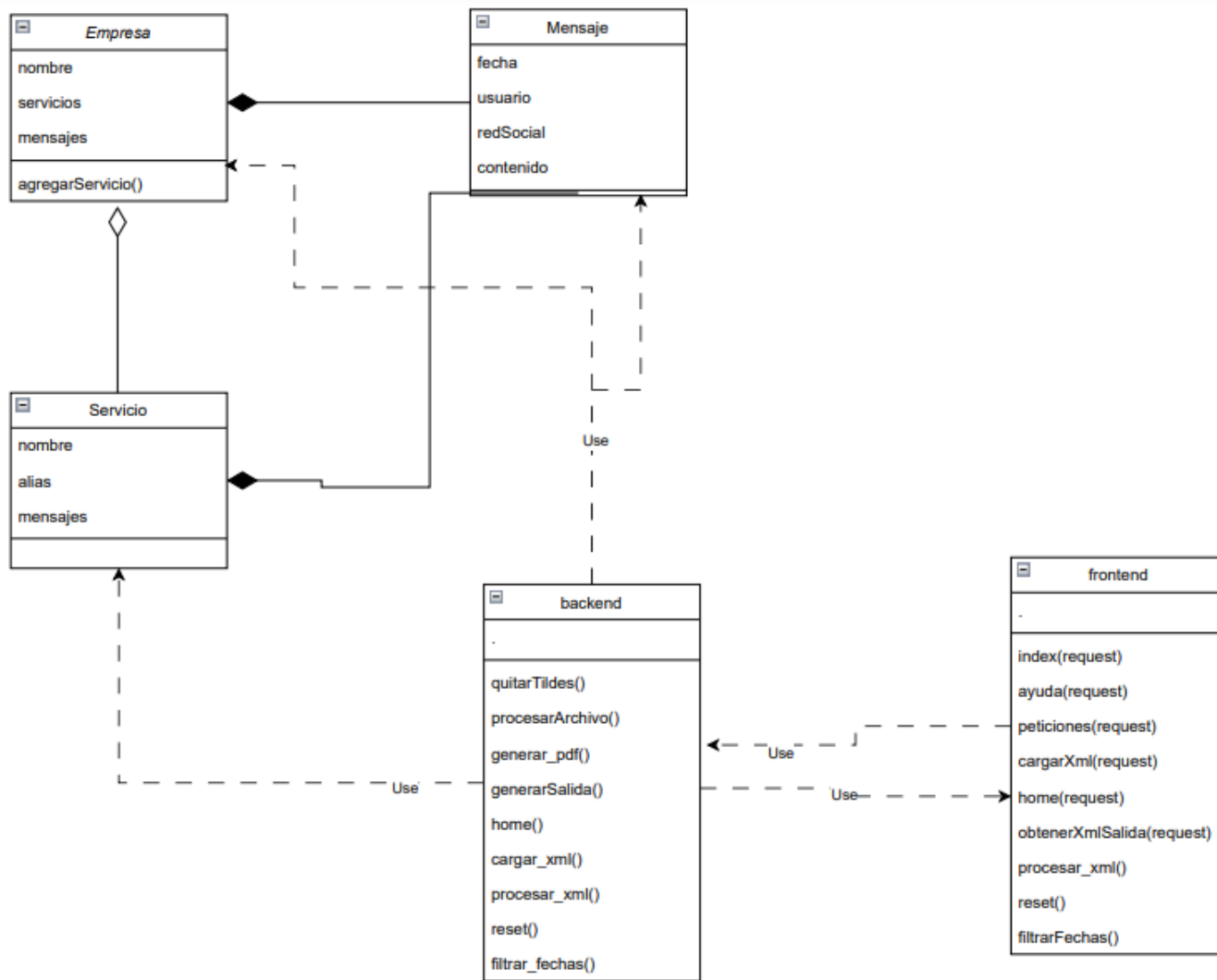


Figura 6 Diagrama de Clases
Fuente: Elaboración Propia