
PROYECTO 3

202010844 – Kevin José de la Cruz Girón

Resumen

La nueva tecnología desarrollada para la red social ChapinChat ofrece una gran mejoría respecto a la capacidad de análisis y definición de perfiles de usuarios que favorecen la identificación de los mismo.

Esta herramienta además de una mejor identificación se utilizan estos datos para brindar un contenido más personalizado a éstos.

Como resultado de este análisis más personalizado se logra una mejoría respecto a un tema de identificación de los perfiles, sobre gustos e intereses.

Se adapta de buena manera en temas de escalabilidad ya que se espera recibir una gran cantidad de mensajes de diferentes y un mismo usuario.

Palabras clave

Xml: Es una especificación de W3C como lenguaje de marcado de propósito general.

Frontend: Se refiere a la parte visible del sitio web que interactúa con los usuarios.

Backend: Se refiere a la parte de un sitio web que maneja los datos y la lógica de negocios detrás de la escena.

Abstract

The new technology developed for the ChapinChat social network offers a great improvement regarding the analysis capacity and definition of user profiles that favor their identification.

This tool, in addition to better identification, uses this data to provide more personalized content to them.

As a result of this more personalized analysis, an improvement is achieved with respect to a subject of identification of profiles, on tastes and interests.

It adapts well to scalability issues since it is expected to receive a large number of messages from different users and the same user.

Keywords

Xml: It is a W3C specification as a general purpose markup language.

Frontend: the part of a web application that is visible to and interacted with by the user.

Backend: refers to the part of the web application that is responsible for processing requests from the frontend and serving the appropriate response..

Introducción

La empresa Tecnologías Chapinas, S.A. ha sido contratada por ChapinChat para desarrollar una herramienta que analice el contenido de la red social y establezca perfiles de sus usuarios con el objetivo de conocerlos mejor y brindar contenido personalizado.

El sistema de información recibirá los mensajes de los usuarios y analizará su contenido para establecer por cada usuario el porcentaje de contenido asociado a cada perfil configurado en el sistema.

Los perfiles se definirán como conjuntos de palabras clave que permitirán calcular el porcentaje de palabras en relación al total de palabras que corresponden a dicho perfil.

La lista de palabras clave será dinámica y podrá cambiar en el tiempo según los intereses de ChapinChat. Además, se define una lista de palabras que no serán consideradas como palabra clave a analizar para lograr una mayor precisión.

Se analizarán las entradas Xml y se cargarán sus datos en archivos xml correspondientes para guardar la data. Como parte del análisis de los perfiles se usan reglas básicas para el cálculo como:

- Las palabras clave de cada mensaje deben ser analizadas y comparadas con la de los perfiles incluyendo la de perfiles descartas.
- Se limpian los mensajes de signos de exclamación , puntuación , entre otros. Posteriormente se compara y se lleva un contador de cada palabra del mensaje con los perfiles establecidos.
- Se hace un promedio de mensajes por usuario respecto a cada perfil para obtener el porcentaje de probabilidad con cada perfil.

- Se retorna al usuario esta información de cuanto % de probabilidad hay de que encaje con cada “X” perfil.

Ejemplo:

Usuario: map001			
Mensaje		% probabilidad perfil “Deportista”	% de probabilidad perfil “Cultura Saludable”
1/4/2023	09:31	10.71%	25%
1/4/2023	12:07	5.65%	40.33%
1/4/2023	15:21	0%	21.5%
1/4/2023	15:33	0%	0
1/4/2023	18:04	7.3%	23.33%
1/4/2023	18:58	21.23%	0%

Para calcular el peso del perfil “Deportista” para el usuario map001 se utilizaría la

siguiente fórmula: $(10.71 + 5.65 + 7.3 + 21.23) / 4 = 11.22$; mientras que, el perfil “Cultura

Saludable” tendría un peso de: $(25 + 40.33 + 21.5 + 23.33) / 4 = 27.54$.

Desarrollo del tema Programación en Python

Python es un lenguaje de scripting independiente de la plataforma y orientado a objetos. Este es un lenguaje de tipo interpretado, esto significa que no es necesario el compilar código fuente para poder realizar su ejecución, esto nos ofrece diversas ventajas como la rapidez de desarrollo e inconvenientes como una menor velocidad. Este es un lenguaje que cuenta con gran popularidad dentro de la comunidad dedicada a la programación y desarrollo de tecnologías esto debido a que Python cuenta con una gran cantidad de librerías, tipos de datos y funciones incorporadas dentro del propio lenguaje, que ayudan a tener una gran agilidad al momento de programar. Además, este es un lenguaje que puede ser considerado sencillo y veloz a comparación de otros como C. A pesar de ser un lenguaje orientado a objetos este también permite el

uso de otros paradigmas de programación, permitiendo que cada programador utilice su propio estilo único o simplemente trabajar como más cómodo se sienta.

XML

Este es un lenguaje de marcado, al igual que el html, definido por el World Wide Web Consortium. El objetivo de este es enfocarse en la simplicidad, generalidad y usabilidad por parte de toda la internet. Este se utiliza para representar estructuras de datos arbitrarias, apuntando a su integración entre sistemas de computadores. Un archivo XML posee dos características principales: marcadores y contenido. Generalmente una secuencia de caracteres iniciará con < y terminará con >. Las secuencias de caracteres que nos son marcadores serán consideradas como “contenido”. Dentro del contenido también pueden existir atributos los cuales están dentro de un tag siendo vistos de esta forma a <nombre>Deportistas </nombre> .

DJANGO

Framework de alto nivel que se enfoca en la eficiencia y la simplicidad del desarrollo de aplicaciones web complejas y escalables. Proporciona una gran cantidad de características y herramientas, incluyendo un ORM (Object-Relational Mapping) para manejar la interacción con la base de datos, un sistema de enrutamiento para manejar las URLs de la aplicación, y un sistema de plantillas para el diseño de páginas web. Django sigue el patrón arquitectónico MVC (Modelo-Vista-Controlador) y proporciona una sólida seguridad integrada y un completo sistema de administración.

promover la reflexión que conduzca a la adopción de una u otra postura.

FLASK

framework minimalista y ligero para el desarrollo de aplicaciones web en Python. Flask no tiene muchos componentes integrados, lo que permite una mayor flexibilidad y personalización en el desarrollo. Flask proporciona un enrutador básico y un sistema de plantillas, pero no tiene una implementación integrada de ORM ni un sistema de administración. Flask es popular entre los desarrolladores que prefieren construir aplicaciones web más pequeñas y ágiles.

Estructura de conexión del sistema

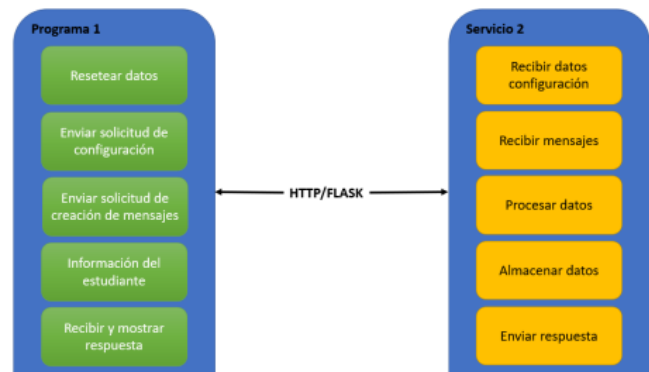


Figura 1. Título o descripción breve de la figura.

Fuente: Universidad de San Carlos

Desarrollo de la aplicación

Para ello se implementó una interfaz con un servidor de flask con la funcionalidad haciendo de backend y un servidor de django funcionando como el frontend.

Implementación de soluciones

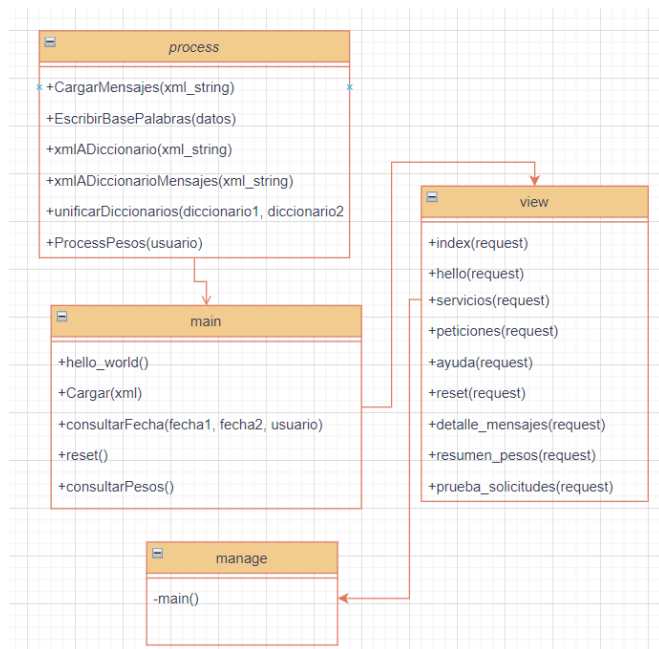
Para el almacenamiento o la base de datos de la aplicación se utilizaron archivos xml, tanto en la lectura como en la carga de estos mismos con la información correspondiente.

Librerías

Se utilizaron varias librerías, para la lectura de xml; como ElementTree, también para la lectura de json y el envío de respuestas de tipo render.

<https://flask.palletsprojects.com/en/2.3.x/>
<https://www.youtube.com/watch?v=T1intZyhXDU>
<https://developer.mozilla.org/es/docs/Learn/Server-side/Django/Introduction>
<https://j2logo.com/tutorial-flask-espanol/>

Diagrama de clases



Conclusiones

Para llegar a esta solución se tuvo que utilizar muchos conceptos y utilidades como Flask, Django, Xml, diccionarios, entre otros. El manejo de archivos en formato Xml fue esencial ya que era la manera de guardar los datos, otra manera de utilizar un tipo de base de datos con archivos pudo haber sido archivos de tipo json, sin embargo con archivos Xml se pudo trabajar de una manera correcta y eficaz para la aplicación.

Referencias bibliográficas

<https://docs.djangoproject.com/en/4.2/>