
ENSAYO PROYECTO 3 IPC2

202307534 – Daniel Alejandro Portillo Garcia

Resumen

Este ensayo habla sobre el desarrollo de una interfaz web usando Python, Flask, y XML para el procesamiento y análisis de archivos XML por medio de comentarios de usuarios. Este programa permite la clasificación y visualización de los mensajes por fecha, compañía, y tipo de comentario, con el fin de ofrecer un análisis de satisfacción sobre los servicios de diversas empresas. El trabajo utiliza tanto herramientas backend como frontend, implementando Python y Flask en el procesamiento y manejo de datos, y HTML y CSS para la interfaz de usuario. Este enfoque no solo es de suma importancia en el mundo de la programación actual web, sino que también permite a los desarrolladores familiarizarse con el uso de APIs y protocolos HTTP para la transmisión y recepción de datos entre servidores y clientes. Este proyecto destaca la funcionalidad y utilidad de emplear estructuras como diccionarios y listas para el almacenamiento de datos para lograr fomentar la comprensión de la programación orientada a objetos en aplicaciones de manejo de datos.

Palabras clave

Desarrollo web, Python, Flask, XML, análisis de datos

Abstract

This essay discusses the development of a web interface using Python, Flask, and XML for processing and analyzing XML files through user comments. This program allows the classification and visualization of messages by date, company, and type of comment, in order to offer a satisfaction analysis of the services of various companies. The work uses both backend and frontend tools, implementing Python and Flask in data processing and management, and HTML and CSS for the user interface. This approach is not only of utmost importance in the world of current web programming, but also allows developers to become familiar with the use of APIs and HTTP protocols for the transmission and reception of data between servers and clients. This project highlights the functionality and utility of using structures such as dictionaries and lists for data storage to promote the understanding of object-oriented programming in data management applications.

Keywords

Web development, Python, Flask, XML, data analysis

Introducción

El desarrollo web es una habilidad indispensable en el área de la programación, donde la habilidad de resolver problemas y adaptar interfaces a las necesidades de los usuarios resulta de suma importancia. En este contexto, la utilización de APIs y protocolos HTTP permite una conexión eficiente entre el frontend y el backend, habilitando el procesamiento y transferencia de datos de forma segura y eficaz. Este proyecto, que utiliza Python y Flask como herramientas principales, se centra en el procesamiento de archivos XML para facilitar el análisis de mensajes de usuarios sobre servicios de distintas empresas, empleando diccionarios y listas para almacenar y organizar los datos de manera accesible y funcional. Este ensayo busca explorar cómo el diseño de esta interfaz puede brindar soluciones prácticas para la gestión de grandes volúmenes de datos, planteando cuestiones como: ¿qué impacto tiene el manejo de datos en la experiencia del usuario?, ¿cómo se integra el análisis de satisfacción en el desarrollo web?

Desarrollo del tema

En el mundo de la programación una de las ramas más importantes es el desarrollo web, puesto que es una de las áreas que se esmera más en poner a prueba las habilidades de ingenieros en sistemas en resolución de problemas que sea adapten de manera intuitiva con los usuarios. En este proyecto se pone a prueba el uso de APIs y protocolos HTTP para el manejo de programación orientada a objetos, con el fin de aprender habilidades de manejo de datos por medio del procesamiento y envío de solicitudes de datos backend y frontend. Con el fin de formar una mejor comprensión de manejo de base de datos por medio de formatos XML.

Para este problema una empresa solicitó el uso de una interfaz que permita el acceso de archivos XML para la lectura de mensajes de usuarios, con el fin de dar un análisis al nivel de aceptación de ciertas empresas, por ello se nos solicita formar un programa web que clasifique repostes XML con la suficiente información para analizar mensajes por su fecha de creación, en la cual se evaluarán la cantidad de comentarios positivos, empresas relacionadas a cada mensaje con sus respectivos servicios para comprender por medio de diccionarios el análisis de la satisfacción que transmite cada comentario, seguidamente formar un reporte resumido para una mejor comprensión del total de resultados de mensajes.

Para esto se optó al manejo de objetos y listas puesto que se pueden almacenar los atributos del XML como formas de matrices, por lo que cada registro formaría una clase que almacenarían los diccionarios, mensajes, empresas, alias, fechas, mensajes, entre otros datos, estos permitirían un acceso de datos más dinámica puesto que se almacenan en listas según atributos, logrando acceder de cualquier momento a los datos, esto simulando una base de datos.

Para las funciones y envíos de todos los métodos se usaron endpoints, que sería las rutas de acceso de comunicación entre funciones del backend y envío de datos del frontend, lo cual permite el acceso, procesamiento y envío de datos para después mostrarlos en la interfaz gráfica del frontend.

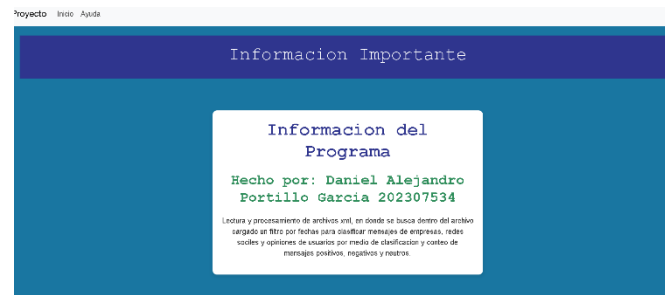
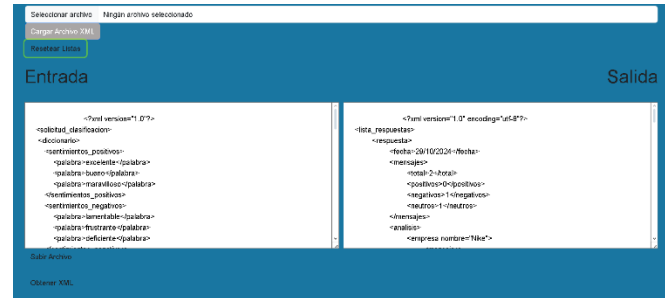
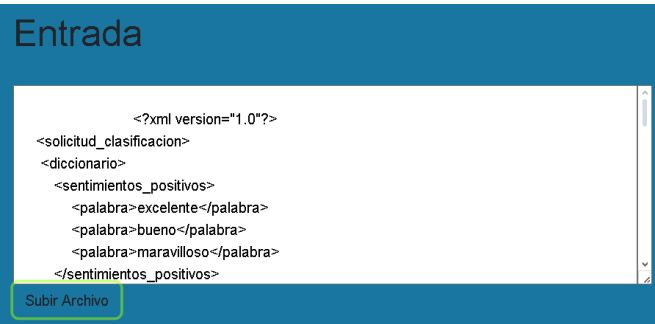
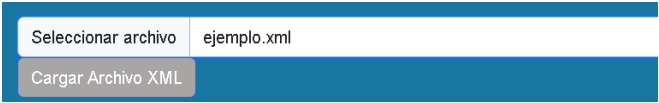
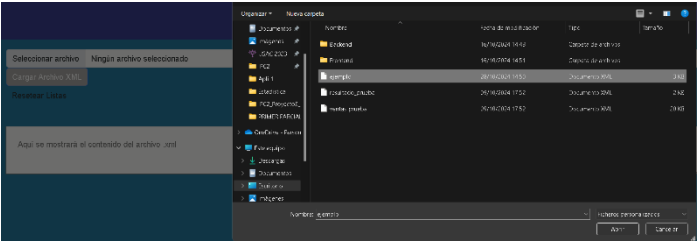
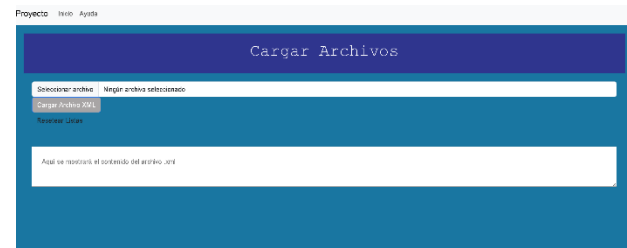
Los formatos de los datos se clasifican de tipo fecha, string y números, string para las palabras reservadas, int para el contador de palabras y tipo fecha para los formatos de día, mes y año de los mensajes adjuntados.

Para la interfaz del usuario se formó una lógica de pasos las cuales se describen de la siguiente manera:

1. Pantalla principal: el usuario tiene opción a subir un archivo XML.
2. El usuario solo podrá seleccionar un archivo XML y se mostrará en un cuadro de texto el archivo seleccionado.
3. Se deberá Subir el archivo para que el frontend haga captura del archivo XML y mostrarlo en pantalla como visa de que se enviara.
4. Enviar el archivo por medio de un botón confirmando su envío y mostrar un mensaje de archivo procesado.
5. Procesar los archivos por medio de un botón que visualizara los archivos procesados en formato XML
- 6.
7. Acceso a un botón para reiniciar la base de datos.
8. Flujo de la barra de navegación para el acceso a la pestaña de ayuda.

Para la ide a utilizar e opto por el lenguaje de programación Python para un mejor manejo y aprendizaje de la temática implementada, como diseño web se usa HTML y CSS. Para las funciones del Backend y fronten se usaron Flask como backkend y Django como comunucacion frontend. Para lectura de Xml minidom y todos los recursos de visualstudio junto a Thunder Client para consultas de Backend.

Flujo del programa:



Conclusiones

- El desarrollo de una interfaz web demuestra cómo las tecnologías modernas pueden ser implementadas para el análisis y visualización de datos con múltiples propósitos útiles para uno o varios usuarios.
- La estructura basada en diccionarios y listas no solo facilita la manipulación de datos, sino que también optimiza el acceso y actualización de los mismos en el tiempo, imitando la funcionalidad de una base de datos sin su forma compleja de manejar datos.
- Se recomienda que futuros ingenieros intenten el desarrollo con bases de datos reales para almacenar datos de manera permanente y ampliar las capacidades de análisis en el ámbito de desarrollo web como se hizo en el proyecto.
- El desarrollo web en la ingeniería permite una práctica real en el uso de herramientas de backend y frontend, fomentando un desarrollo de resolución mas intuitivo para las personas a los que los ingenieros asistan en futuros proyectos

Referencias bibliográficas

Ronacher, A. (2023). Flask (versión 2.3.3) [Framework]. <https://flask.palletsprojects.com>

Django Software Foundation. (2023). Django (versión 4.2.5) [Framework]. <https://www.djangoproject.com>

Python Software Foundation. (2023). Python (versión 3.11.5) [Software]. <https://www.python.org>

Microsoft Corporation. (2023). Visual Studio (versión 17.7) [Software]. <https://visualstudio.microsoft.com>

Anexos

Diagrama de Clases

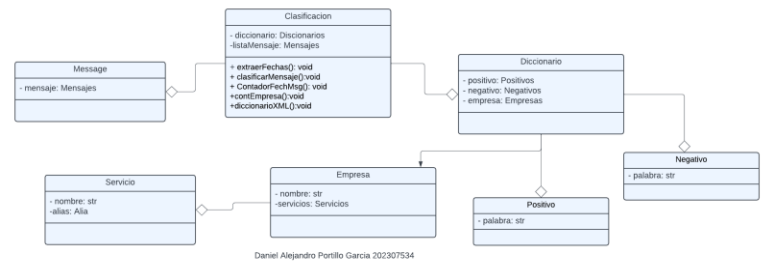


Diagrama de actividades

