Proyecto 3. Flask with Django

201905515 – CARLOS AUGUSTO CALDERON ESTRADA

Resumen

Se desarrollo una aplicación web (backend/frontend) encargada de recibir un archivo XML por parte del frontend, y este enviarlo al backend para posteriormente ser modificado y generar un archivo nuevo. Una vez con el archivo nuevo generado se puede acceder a dos tipos de gráficas, una para resumen de IVA y otra de resumen por fechas, terminando con el funcionamiento de la aplicación. Uno de los puntos más importantes fue la comunicación entre el backend y frontend ya que constantemente estuvieron enviando peticiones uno al otro.

Palabras clave
XML, BACKEND, FRONTEND

Abstract

A web application (backend/frontend) was developed to receive an XML file from the frontend and send it to the backend to be later modified and generate a new file. Once the new file is generated, two types of graphs can be accessed, one of IVA and the other of general requests, finishing with the operation of the application. One of the most important points was the communication between the backend and frontend as they were constantly sending requests to each other.

Keywords
XML, BACKEND, FRONTEND

Introducción

Las aplicaciones web son cada vez más solicitadas por las empresas o clientes en general, por lo mismo el desarrollo de este proyecto permite comprender como es el funcionamiento y la comunicación entre el backend y el frontend. Trabajando con django para frontend y flask para backend, se logró una comunicación eficiente entre ambos, haciendo peticiones el uno al otro para lograr el correcto funcionamiento de la aplicación.

Desarrollo del tema

El principal funcionamiento de la aplicación web consisten en enviar un archivo XML desde el frontend hacia el backend, una vez recibido el archivo, se procede a trabajarlo mediante arreglos o listas que nos permite utilizar Python, terminado el nuevo archivo de salida, se crea, se guarda y se envía de nuevo al frontend para que pueda ser mostrado en su respectiva área de texto. Otra de las funcionalidades es que se pueden crear dos tipos de gráficas, permitiéndonos así, observar el correcto filtrado de información del archivo original.

Las aplicaciones web son un tipo de software que se codifica en un lenguaje soportado por los navegadores web y cuya ejecución es llevada a cabo por el navegador en Internet o de una intranet (de ahí que reciban el nombre de App web).

Otra definición que podríamos dar para una aplicación web es la siguiente:

Son aquellas herramientas que los usuarios pueden utilizar accediendo a un servidor web por medio de Internet o de una intranet mediante un navegador que ejecutará la misma.

¿Cómo funcionan las Apps web?

Que las aplicaciones web sean ejecutadas por medio de un navegador web en una red significa que los datos o los archivos en los que trabajas son procesados y almacenados dentro de la una red a través de un navegador. Por este motivo, este tipo de aplicaciones por lo general, no necesitan ser instaladas en el ordenador o el móvil.

Una página Web puede contener elementos que permiten una comunicación activa entre el usuario y la información, haciendo que éste acceda a los datos de forma interactiva, ya que el sitio web se encargará de responder a cada una de las acciones que éste ejecute (por ejemplo acceder a gestores de bases de datos de todo tipo, publicar e interactuar con los contenidos, rellenar y enviar formularios, participar en juegos, etc.

Las aplicaciones web están íntimamente relacionadas con el almacenamiento de datos en la nube, ya que toda la información se guarda de forma permanente en servidores web, los cuales además de alojar dicha información, nos la envían a nuestros dispositivos móviles o equipos informáticos en cada momento que sea requerida, realizando copias temporales de estos envíos dentro de los equipos y dispositivos que utilicemos.

¿Por qué son tan populares las aplicaciones web?

Las aplicaciones web son muy populares debido a:

- La practicidad que ofrecen los navegadores web como clientes ligeros. Un cliente ligero (Thin Client) es un sistema que trabaja en una arquitectura de red cliente-servidor en la cual existe muy poca o ninguna lógica del programa,

por lo que depende principalmente del servidor central para las tareas de procesamiento.

- La independencia del sistema operativo que uses en tu ordenador o dispositivo móvil.
- La facilidad para actualizar y mantener aplicaciones web sin la necesidad de tener que distribuir el software o que se tengan que instalar el mismo por los usuarios potenciales.
- El libre acceso de los usuarios en cualquier momento, lugar o dispositivo, sólo con tener conexión a Internet y los datos de acceso (nombre usuario y contraseña).

Tipos de Apps web que se pueden desarrollar

Existen miles de páginas y aplicaciones web asociadas. A continuación, se presentan algunos ejemplos de aplicaciones y diseño web que se pueden llegar a desarrollar:

- Web mail: Sistemas de acceso al correo electrónico que permiten acceder a tus correos mediante un navegador web, sin tener que descargar los propios correos en el ordenador. Para ello utilizan clientes del tipo Gmail, Outlook, etc
- Wikis: Sitios y aplicaciones web cuyas páginas y contenidos pueden ser editados directamente desde el navegador, donde los usuarios crean, modifican o eliminan contenidos que, generalmente comparten.
- Weblogs: Sitios y aplicaciones web cuyas páginas y contenidos son de fácil actualización, de tal que permite a sus autores publicar contenidos (textos, imágenes y otros archivos) con aparentar un solo botón, ya que suelen contar con un editor y herramientas para tal efecto en la propia web.
- Tiendas Online: Tipos de comercio que usan como medio principal para realizar sus transacciones un sitio web y/o una aplicación conectada a internet desde la que los usuarios y clientes pueden realizar sus compras.

Ventajas de las aplicaciones web

Las ventajas más importantes que tiene el desarrollo de una App web son las siguientes:

- Ahorro de tiempo: Son Apps sencillas de gestionar, por lo que permiten realizar tareas de forma fácil sin necesidad de descargar ni instalar ningún programa o plugin adicional.
- Completa compatibilidad: Son totalmente compatibles con navegadores para poder utilizarlas. Sólo se suele requerir que el navegador web esté debidamente actualizado para poder usar este tipo de Apps.
- Actualización continua e inmediata: Debido a que es el propio desarrollador App el que gestiona y controla el software, la versión que descarguen, instalen y utilicen los usuarios, siempre será la última que haya lanzado dicho desarrollador App. Para ello es imprescindible estar al tanto de lo que ocurre con la App.
- Recuperación de datos: Una de las mayores ventajas de una App Web es que, en la mayoría de las ocasiones el usuario puede recuperar su información desde cualquier dispositivo y lugar con su nombre de usuario y contraseña.

- Ahorro de recursos en equipos y dispositivos: Las Apps Web, generalmente tiene un bajo consumo de recursos dado que toda (o gran parte) de la aplicación se encuentra en un servidor web y no en nuestro ordenador.

a.**Django**

Django es un framework de desarrollo web de código abierto, escrito en Python, que respeta el patrón de diseño conocido como modelo–vista–controlador (MVC). Fue desarrollado en origen para gestionar varias páginas orientadas a noticias de la World Company de Lawrence, Kansas, y fue liberada al público bajo una licencia BSD en julio de 2005; el framework fue nombrado en alusión al guitarrista de jazz gitano Django Reinhardt.

En junio de 2008 fue anunciado que la recién formada Django Software Foundation se haría cargo de Django en el futuro.

La meta fundamental de Django es facilitar la creación de sitios web complejos. Django pone énfasis en el re-uso, la conectividad y extensibilidad de componentes, el desarrollo rápido y el principio No te repitas (DRY, del inglés Don't Repeat Yourself). Python es usado en todas las partes del framework, incluso en configuraciones, archivos, y en los modelos de datos.

b. **Flask:**

Flask es un “micro” Framework escrito en Python y concebido para facilitar el desarrollo de Aplicaciones Web bajo el patrón MVC.

La palabra “micro” no designa a que sea un proyecto pequeño o que nos permita hacer páginas web pequeñas, sino que al instalar Flask tenemos las herramientas necesarias para crear una aplicación web funcional, pero si se necesita en algún momento una nueva funcionalidad hay un conjunto muy grande extensiones (plugins) que se pueden instalar con Flask que le van dotando de funcionalidad.

De principio en la instalación no se tienen todas las funcionalidades que se pueden necesitar, pero de una manera muy sencilla se pueden extender el proyecto con nuevas funcionalidades por medio de plugins.

El patrón MVC es una manera o una forma de trabajar que permite diferenciar y separar lo que es el modelo de datos (los datos que van a tener la App que normalmente están guardados en BD), la vista (página HTML) y el controlador (donde se gestiona las peticiones de la app web).

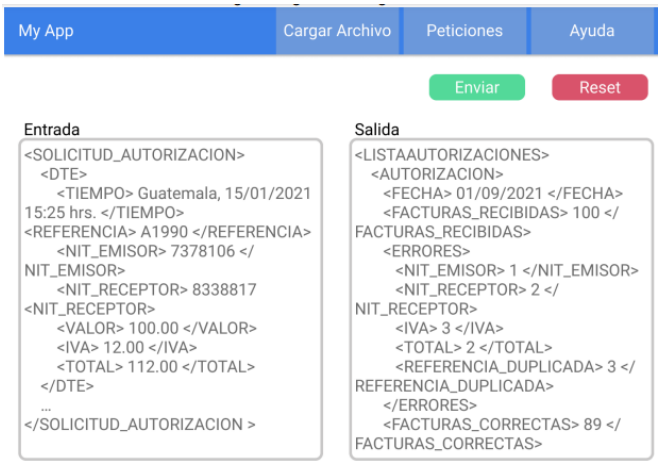

Figura 1. Interfaz de inicio.

Fuente: Instrucciones del proyecto

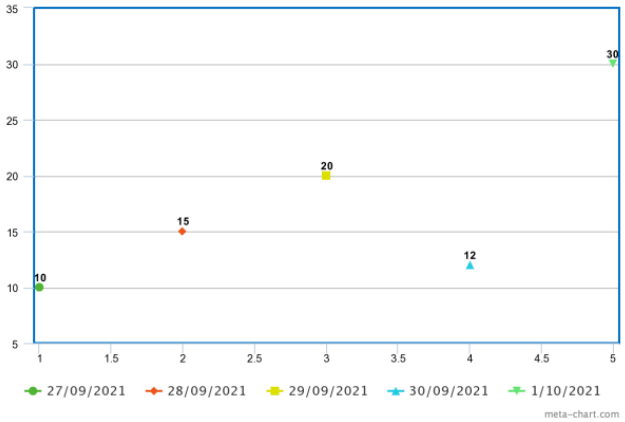

Figura 2. Interfaz de la grafica

Fuente: Instrucciones del proyecto

```
<SOLICITUD_AUTORIZACION>
  <DTE>
    <TIEMPO> Guatemala, 15/01/2021 15:25 hrs. </TIEMPO>
    <REFERENCIA> A1990 </REFERENCIA>
    <NIT_EMISOR> 7378106 </NIT_EMISOR>
    <NIT_RECEPTOR> 8338817 <NIT_RECEPTOR>
    <VALOR> 100.00 </VALOR>
    <IVA> 12.00 </IVA>
    <TOTAL> 112.00 </TOTAL>
  </DTE>
  ...
</SOLICITUD_AUTORIZACION >
```

Figura 3. Archivo de entrada SolicitudDTE XML

Fuente: Instrucciones del proyecto

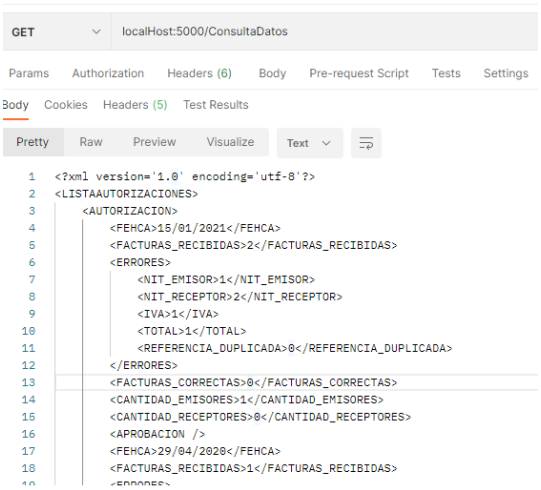

Figura 4. Archivo de salida XML desde Postman

Fuente: elaboración Propia

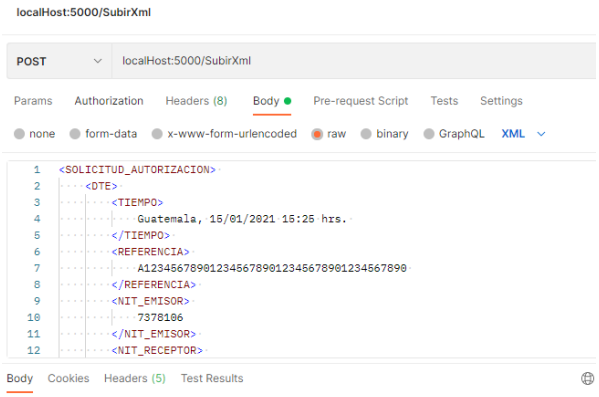

Figura 5. Endpoint de entrada XML desde Postman

Fuente: elaboración Propia

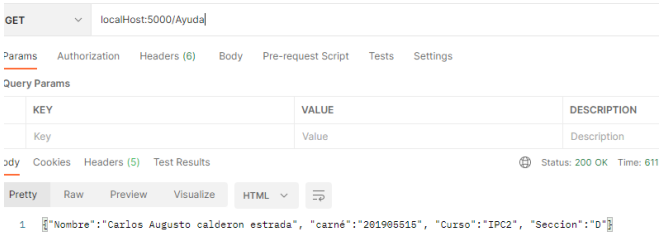

Figura 6. Endpoint de ayuda desde postman

Fuente: elaboración propia

Conclusiones

Encontramos diferentes tipos de ventajas al momento de utilizar una aplicación web, por lo mismo, estas son muy solicitadas, entonces el correcto uso de un buen framework como Django, nos permite el eficiente desarrollo y fácil mantenimiento de la misma.

La comunicación entre ambos servidores tiene que desarrollarse de manera correcta, de no ser así, el funcionamiento de la aplicación no seria correcto y se estaría trabajando de manera ineficiente.

Referencias bibliográficas

- Anonimo, (2021). *¿Qué son las aplicaciones web?. Disponible en:* ¿Qué son las Aplicaciones Web? Ventajas y Tipos de Desarrollo Web (wiboomedia.com)
- Anonimo, (2021). *Django (framework). Disponible en:* Django (framework) - Wikipedia, la enciclopedia libre
- Anonimo, (2021). Qué es Flask. *Disponible en:* Qué es Flask y ventajas que ofrece | OpenWebinars
- Anonimo, (2021). *Tipo de dato abstracto. Disponible en:*https://es.wikipedia.org/wiki/Tipo_de_dato_abstracto
- Anonimo, (2021). *Tipo de dato abstracto. Disponible en:* https://es.wikipedia.org/wiki/Matriz_(matem%C3%A1ticas)