
PROYECTO 3

202004765 – Javier Alejandro Gutierrez de León

Resumen

Utilizando el lenguaje de programación Python se desarrolló una solución para la Superintendencia de Administración Tributaria para solicitar la autorización de un Documento Tributario Electrónico (DTE) implementando un API (utilizando Flask) que brinda servicios utilizando el Protocolo HTTP bajo el concepto de programación orientada a objetos (POO) a un programa base desarrollado utilizando Django con el cual se le pueden hacer las consultas a la API, la cual posee una base de datos creada con XML logrando de esta forma retornar los datos que el programa principal solicita, mostrándolos en tablas y generando graficas utilizando otra API llamada 'QuickChart' con el fin de mostrar de una manera más agradable y entendible los datos al usuario final.

Palabras clave

HTTP: Protocolo de transferencia de hipertexto que permite transferencias de información a través de archivos en la World Wide Web.

API: es un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones.

Framework: conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular

Abstract

Using the Python programming language, a solution was developed for the Tax Administration Superintendency to request the authorization of an Electronic Tax Document (DTE) implementing an API (using Flask) that provides services using the HTTP Protocol under the concept of object-oriented programming. (OOP) to a base program developed using Django with which you can make queries to the API, which has a database created with XML, thus managing to return the data that the main program requests, showing them in tables and generating graphs using another API called 'QuickChart' in order to show the data in a more pleasant and understandable way to the end user.

Keywords

HTTP: Hypertext Transfer Protocol that allows information transfers through files on the World Wide Web.

API: is a set of definitions and protocols used to develop and integrate application software.

Framework: standardized set of concepts, practices and criteria to address a particular type of problem.

Introducción

La Superintendencia de Administración Tributaria le ha solicitado construir un software que pueda ser consumido desde Internet como un servicio. Este software recibirá un mensaje con los datos para solicitar la autorización de un Documento Tributario Electrónico (DTE) emitido por un contribuyente y como respuesta emitirá un número único de autorización, este número será un correlativo que iniciará con el valor 1 cada día y no deberá repetirse de nuevo en ese día.

Desarrollo del tema

Para el desarrollo de este proyecto fue necesario utilizar diferentes herramientas que se complementan entre sí para lograr darle solución al problema propuesto, entre las cuales podemos destacar:

Expresiones Regulares: son patrones que se utilizan para hacer coincidir combinaciones de caracteres en cadenas.

XML: es un lenguaje de marcado similar a HTML. Significa Extensible Markup Language (Lenguaje de Marcado Extensible) y es una especificación de W3C como lenguaje de marcado de propósito general. ... El propósito principal del lenguaje es compartir datos a través de diferentes sistemas, como Internet.

HTML: (Lenguaje de Marcas de Hipertexto, del inglés HyperText Markup Language) es el componente más básico de la Web. Define el significado y la estructura del contenido web.

CSS: (Hojas de estilo en cascada), es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado.

Hilo: es simplemente una tarea que puede ser ejecutada al mismo tiempo que otra tarea.

API: es un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones. API significa interfaz de programación de aplicaciones. Las API permiten que sus productos y servicios se comuniquen con otros, sin necesidad de saber cómo están implementados.

Framework: Es un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar

Request: son métodos predeterminados en nuestro servidor (HTTP), o sea son funciones privadas del servidor que tengas instalado y los cuales devuelven una variable en formato arreglo asociativo, entre los principales métodos tenemos:

GET: solicita una representación de un recurso específico. Las peticiones que usan el método GET sólo deben recuperar datos.

HEAD: pide una respuesta idéntica a la de una petición GET, pero sin el cuerpo de la respuesta.

POST: se utiliza para enviar una entidad a un recurso en específico, causando a menudo un cambio en el estado o efectos secundarios en el servidor.

PUT: reemplaza todas las representaciones actuales del recurso de destino con la carga útil de la petición.

DELETE: borra un recurso en específico.

CONNECT: establece un túnel hacia el servidor identificado por el recurso.

OPTIONS: es utilizado para describir las opciones de comunicación para el recurso de destino.

TRACE: realiza una prueba de bucle de retorno de mensaje a lo largo de la ruta al recurso de destino.

PATCH: es utilizado para aplicar modificaciones parciales a un recurso.

Conclusiones

Utilizar el lenguaje Python en ambos servicios fue de gran ayuda, ya que al ser un lenguaje mas simple se lograron realizar funciones complejas en pocas líneas de código, las cuales hubieran sido muchas mas si se hubiese utilizado algún otro lenguaje de programación.

La manipulación de archivos XML por medio de cElementTree fue muy sencillo ya que con las funciones que este posee fue fácil extraer la información dentro del archivo, ahorrando la interpretación de línea por línea y la eliminación de caracteres que no fueran relevantes que se debería realizar si no utilizáramos esta librería.

La utilización de Graphviz fue QuickChart fue de gran utilidad ya que con los mismos parámetros obtenidos de la respuesta de la API (Servicio 2) se podían generar graficas complejas que hubieran tenido gran complejidad de elaborar si no se hubiese utilizado esta API.

Referencias bibliográficas

- Aymar, S., Barahona, G., Gomez, M., & Martinez, C. (2019). INTELIGENCIA ARTIFICIAL (BÚSQUEDA A ESTRELLA). UNIVERSIDAD MAYOR DE SAN SIMÓN FACULTAD DE CIENCIAS Y TECNOLOGÍA. http://www.cs.umss.edu.bo/doc/material/mat_gral_139/Busqueda%20A%20estrella-2.pdfMuniz
- [Estructura de datos] -Lista doblemente enlazada, use Python para crear una lista simple doblemente enlazada (explicación detallada) - programador clic. (s. f.). Programmerclick. Recuperado 16 de agosto de 2021, de <https://programmerclick.com/article/2000768743/>
- F. T., & Severance, C. R. (2015). Python para informaticos: Explorando la informacion. Createspace Independent Publishing Platform.
- Graphviz. (s. f.). Graphviz. Recuperado 15 de agosto de 2021, de <https://graphviz.org/>
- xml.etree.ElementTree — La API XML de ElementTree — documentación de Python - 3.9.6. (s. f.). docs.Python. Recuperado 16 de agosto de 2021, de <https://docs.python.org/es/3/library/xml.etree.elementtree.html>

Anexos

