

---

## Proyecto 3

---

202004816 - Kevin Steve Martinez Lemus

### Resumen

Con el objetivo de dar solución al problema planteado, utilizando la lógica y conocimientos adquiridos durante el curso de Introducción a la Programación y Computación 2, implementando el lenguaje de programación Python y utilizando el entorno de trabajo de Visual Studio Code. Se realizó un programa el cual realiza diferentes procesos para la validación y autorización de Documentos Tributarios Electrónicos (DTE) en la SAT. Leyendo archivos XML como entrada y generando archivos XML de salida, también generando gráficas y resúmenes por fecha, así como un reporte PDF. Para finalizar, el proyecto fue un éxito ya que este válida cada parte de las facturas para generar un código de aprobación y así tener en orden todas las facturas aprobadas.

### Palabras clave

- **Servidor BackEnd:** es la parte o rama del desarrollo web encargada de que toda la lógica de una página funcione. Consiste en el conjunto de acciones que pasan dentro de una web, pero que no podemos ver.
- **Interfaz:** es la parte del desarrollo web que se dedica a la parte frontal de un sitio web, en pocas palabras del diseño de un sitio web, desde la estructura del sitio hasta los estilos como colores, fondos, tamaños hasta llegar a las animaciones y efectos.
- **Petición:** es un módulo de Python que puedes usar para enviar todo tipo de peticiones HTTP.
- **XML:** Es un lenguaje de marcado similar a HTML. Significa Extensible Markup

Language (Lenguaje de Mercado Extensible).

## **Abstract**

*With the aim of solving the problem, using the logic and knowledge acquired during the Introduction to Programming and Computing 2 course, implementing the Python programming language and using the Visual Studio Code working environment. A program was carried out which carries out different processes for the validation and authorization of Electronic Tax Documents (DTE) in the SAT. Reading XML files as input and generating XML files as output, also generating charts and summaries by date, as well as a PDF report. Finally, the project was a success since each part of the invoices is valid to generate an approval code and thus have all the approved invoices in order.*

## **Keywords**

• **BackEnd Server:** It is the part or branch of web development in charge of making all the logic of a page work. It consists of the set of actions that happen within a web, but that we cannot see.

• **FrontEnd:** It is the part of web development that is dedicated to the front part of a website, in short of the design of a website, from the structure of the site to the styles such as colors, backgrounds, sizes to the animations and effects.

• **Request:** it is a Python module that you can use to send all kinds of HTTP requests.

• **XML:** It is a markup language similar to HTML. It stands for Extensible Markup Language.

## **Introducción**

Este ensayo tiene el fin de explicar a detalle el problema planteado, así como la solución propuesta en forma de aplicación para la validación y autorización de Documentos Tributarios Electrónicos (DTE) en la Superintendencia de Administración Tributaria (SAT). Esta aplicación cuenta con una página web para facilitar la interacción entre usuario y aplicación, y un backend para el guardado de los datos.

## Aplicación

El software que plantea la Superintendencia de Administración Tributaria es el siguiente:

“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. La estructura del número de autorización es la siguiente: `yyyymmdd#####`.”

El programa por desarrollar, luego de recibir el mensaje antes mencionado, deberá almacenar la información necesaria en un archivo de salida XML, para posteriormente emitir reportes y realizar consultas.”

Para generar una solución viable a este problema, lo primero que se realizó fue un diagrama de clases el cual nos describe a detalle, gráficamente la estructura de backend que se implementará en nuestra aplicación utilizando el famoso paradigma de Programación Orientada a Objetos (POO), el diagrama realizado es el siguiente:

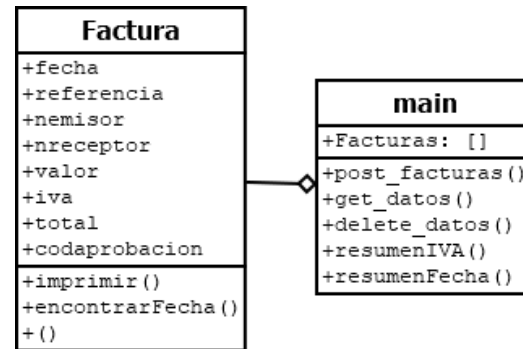


Figura 1. Diagrama de Clases

Fuente: Elaboración Propia 2021

Esta aplicación fue programada con uno de los lenguajes de programación multiparadigma conocido como Python, y realizado en el entorno de trabajo de Visual Studio Code. Utilizando para el backend la librería de flask para poder hacer las peticiones necesarias y la librería ElementTree para leer el archivo de entrada XML, mientras que, para el frontend se utilizó django que es un framework de desarrollo web de código abierto, escrito en Python, que respeta el patrón de diseño conocido como modelo–vista–controlador, en donde, se utilizó html para toda la interfaz.

Las opciones de este sistema se ubican en la parte superior de la página web, en la barra de navegación, en ellas se puede : consultar datos, generar los diferentes resúmenes implementados (Resumen de IVA por fecha y NIT y Resumen por rango de fechas), generar sus respectivas gráficas, y generación del reporte PDF. Para poder ingresar los

datos se necesita de un archivo xml o bien, se puede escribir en el apartado de entrada dentro de la página, luego de tener el archivo que se desea analizar, para poder analizarlo existe el botón de enviar, que extrae el texto ingresado y lo envía al backend para analizarlo, también existe la opción de reset la cual elimina todos los datos almacenados para empezar desde cero.

Para realizar las gráficas correspondientes a los resúmenes, se utilizó la herramienta de matplotlib, esta es una librería de Python especializada en la creación de gráficos en dos dimensiones, y permite crear y personalizar los tipos de gráficos más comunes. Para la creación del reporte en PDF, se utilizó la librería de reportlab que nos permite realizar documentos PDF. Ya finalizado el programa, este se sometió a una serie de pruebas para corroborar su buen funcionamiento, obteniendo resultados satisfactorios. Este proyecto se subirá a un repositorio de GitHub en la nube.

## Conclusiones

- El uso del paradigma de Programación Orientada a Objetos (POO) facilita la programación de soluciones de esta índole.

- La solución planteada se acepta ya que esta aplicación web válida las facturas recibidas y las ordena de la mejor manera.

## Bibliografía

- MDN Web Docs. (2021). Obtenido de [https://developer.mozilla.org/es/docs/Web/XML/XML\\_introduction](https://developer.mozilla.org/es/docs/Web/XML/XML_introduction)
- Stefaniak, P. (26 de 07 de 2019). *descubre comunicación*. Obtenido de <https://descubrecomunicacion.com/que-es-backend-y-frontend/>