
TRIBUTARYADMINISTRATION

202001534 – Erwin Fernando Vásquez Peñate

Resumen

Las bases de datos son conjuntos de datos fundamentales conformados con mucha información, las anteriormente mencionadas son un conjunto de secuencias que guardan datos de manera arbitraria, los cuáles son referenciados mediante atributos y claves, para fines del proyecto, las bases de datos fueron implementadas en archivos xml.

Los archivos de texto plano como por ejemplo los xml, últimamente son muy utilizados y esto se debe a que su estructura es muy amigable tanto como para lectura, como para implementar Programación orientada a objetos (POO).

En términos generales la necesidad de ver gráficamente lo que ocurre en nuestras aplicaciones es muy importante, ya que de esta manera es mucho más fácil entenderlo y poder ayudar a los demás usuarios a entenderlo, por eso la implementación de herramientas como reportes es innovador e indispensable a la vez.

Dadas las necesidades anteriores y el impacto en la programación es necesaria la implementación conjunta de las herramientas mencionadas.

Palabras clave

- Django
- Autómata
- Estructura Xml
- Flask

Abstract

The databases are fundamental data sets made up of a lot of information, the aforementioned are a set of sequences that arbitrarily save data, which are referenced by attributes and keys, for the purposes of the project, the databases were implemented in xml files.

The text plain files, like xml, in the present are very used by the developer community, because these files are very helpful, with storing information and be able to read it easily, in addition they are a good example of Object-oriented programming (POO).

In the abstract applications is necessary to control application flow, so with this need, we turn to alternative options as support tools, like reports and with this, we can observe in a graphic way the operation of our work.

Given the above needs and the impact on programming, the joint implementation of the aforementioned tools is necessary.

Keywords

- Django
- Automaton
- Xml Structure
- Flask

Introducción

Conociendo la importancia de diversas herramientas para la creación de aplicaciones muy abstractas, podemos adentrarnos a un mundo nuevo y lleno de incógnitas que mediante investigación y sobre todo mucha práctica pueden ser respondidas de una manera muy dócil, basándonos en el problema general, nos encontramos en una situación donde una empresa necesita mediante un algoritmo poder lograr una comunicación entre un sitio web levantado mediante django y la api, que es manejada con el framework de flask, los cuales mediante peticiones deben comunicarse entre sí y que todo pueda ser llevado a cabo de una manera satisfactoria y ahorrando la mayor cantidad de trabajo posible comparando el traslado de información de forma manual.

Desarrollo del tema

Al inicio podemos encontrarnos perdidos dentro de un mundo nuevo lleno de incógnitas, pero mediante el adentramiento a este “nuevo mundo” llamado programación podemos darnos cuenta que siempre se aprenden nuevas cosas y nuevas habilidades, que seguramente antes no teníamos.

Hablando un poco del problema central de la aplicación, necesitamos tener 2 cosas en la mente, la primera es que tiene que ser una aplicación amigable para el usuario por lo tanto debeos contar con un menú interactivo y fácil de utilizar, mientras más fácil sea de utilizar para el usuario, será mucho mejor la repercusión y aceptación que pueda tener el proyecto, asimismo se necesita una muy buena vista en la interfaz gráfica de la aplicación, para lo cuál se usará tanto CSS como Bootstrap , con lo cual se le puede dar una muy buena utilidad a todas sus funciones que ofrece.

Al implementar herramientas como archivos con extensión “Xml”, no solo tenemos que estar preparados para su posible lectura y escritura, sino que también conocer cada una de sus características, y en este proyecto hablar de los archivos xml es dar con “la punta del icebeg”, pues mediante el conocimiento de su estructura podemos darnos cuenta, que un archivo “Xml” no es más que una jerarquía sintáctica de objetos, ya que mientras más nos vallamos sumergiendo en el archivo encontraremos distintas ramas, las cuales pueden llamarse “nodos”.

Para adentrarnos un poco más a la estructura de la aplicación se muestra un posible prototipo para el menú principal de la aplicación y posteriormente pueden exponerse cada una de las funciones.



Figura 1. Prototipo menú principal.

Fuente: Elaboración propia, 2021

Según se vea el problema inicial se puede intentar resolver de muchas formas, pero muchas de ellas son poco efectivas y mucho más complejas, para ello hay que entender con que se está tratando.

Teniendo un prototipo del menú podemos darle un punto de partida a nuestro flujo de aplicación, empezando por lo menos abstracto y dejar lo más complejo para evitar la pérdida innecesaria de tiempo e ideas.

a. Manejo de archivos Xml

La lectura del “xml” podemos adelantarnos a pensar que puede ser muy complicada y compleja pero la realidad es otra ya que con distintas librerías del lenguaje Python podemos leerlos e incluso escribir archivos de esta naturaleza en cuestión de minutos, a continuación, se presentan algunas recomendaciones de librerías para la lectura y escritura de archivos “xml”.

Tabla I.

Librería para Lectura “XML”

FROM	IMPORT
Xml.dom	minidom

Tabla II.

Librería para Escritura “XML”

IMPORT	AS
Xml.etree.ElementTree	ET

Con la lectura del “xml”, ya podemos poner en marcha lo más importante, y es que las bases de datos están vacías y necesitan llenarse de objetos de nombre DTE, cada objeto tendrá como atributo un nit, referencia, iva, total, etc.

Con las bases de datos simuladas en los archivos xml nuestra información siempre estará conectada y eso es fundamental para todas las validaciones que tendremos que realizar en todo momento, por ejemplo, validar que el DTE que se quiere procesar exista y sea válido, si no se acepta el DTE, la aplicación mostrará un aviso y se le solicitará uno nuevo hasta que se valide mediante las bases de datos y cada uno de sus registros que el DTE solicitado exista.

b. Cargar Archivo

En esta sección del menú principal se mostrará una opción en la cual el usuario puede cargar un archivo de entrada con extensión “xml”, donde al cargarlo y darle al botón enviar se puede como su nombre lo indica mandar la información a backend para que pueda ser validada y posteriormente procesada.

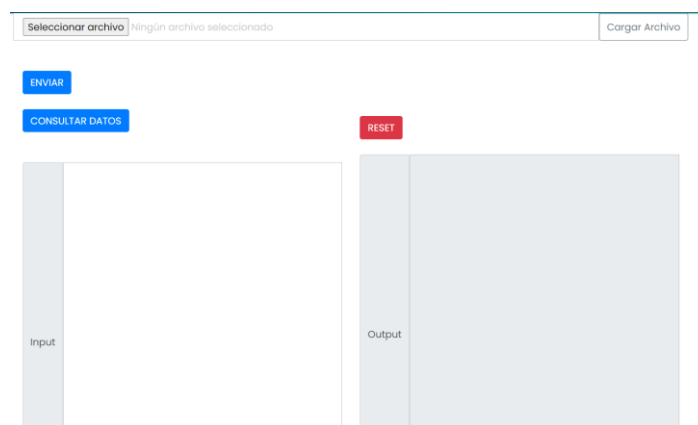


Figura 2. Cargar Archivo.

Fuente: Elaboración propia, 2021

Luego de enviar, también se pueden hacer e interactuar con las demás peticiones, recordando que se trabajará solamente con los datos enviados a backend que fueron procesados, con esto en mente ya podemos pasar a ver las diferentes peticiones que se pueden realizar.

Las bases de datos simuladas pueden encontrarse y verse de una manera eficiente, aunque en muchos casos las bases de datos simuladas en archivos xml no son suficientes y se tiene que recurrir a una base de datos como tal, para todas las funciones necesarias.

Una ventaja impresionante de usar estos métodos como las bases de datos simuladas, es que no te acostumbrarás a usar las funciones que ya vienen establecidas en los lenguajes (Python, en este proyecto), como métodos ya predefinidos como append, sort, pop, remove, etc.

c. Datos de Estudiante

La quinta opción del menú, donde mediante un pequeño apartado se pueden corroborar los datos del estudiante que diseñó la aplicación, y también se indica el curso y sección al cual pertenece, además de poder descargar un mini manual para el uso de la aplicación.



Documentación

Ensayo.pdf

Figura 3. Datos estudiantiles.

Fuente: elaboración propia, 2021.

d. Peticiones

La aplicación cuenta con algunas peticiones disponibles para poder ser realizadas, como desplegar la información por rango de fechas o una fecha en específico. Además de generar reportes mediante gráficas y pdf.



Figura 4. Peticiones.

Fuente: elaboración propia, 2021.

Conclusiones

- En la actualidad es muy necesario incluir en las aplicaciones herramientas que permitan ver de otra forma el flujo, en este caso hablamos de una forma gráfica, y por eso la generación de gráficas y reportes pdf es una muy buena opción.
- Los archivos xml son universales dentro de la programación, por lo mismo los lenguajes se ven en la necesidad de implementar librerías para su lectura y escritura.
- Las bases de datos simuladas son una muy buena forma de almacenar información de manera ordenada, y donde cada uno de sus registros va conectado y gracias a distintos ciclos se pueden agregar, eliminar o actualizar datos de los distintos registros.

Referencias bibliográficas

- M.E.S College of Engineering, (2006), Principles of data Structures using C and C++.
- School of Computer Science, University of Birmingham, UK, (2018), Data Structures and Algorithms.