
REPRODUCTOR DE MÚSICA ONLINE IPCMUSIC

201603028 – Oscar Eduardo Morales Girón

Resumen

IPCMusica es un reproductor de música online basado en el lenguaje de programación Python, su funcionamiento se basa en la utilización de dos servicios el Frontend y Backend, el Frontend se encarga de facilitar al usuario el uso de la aplicación que implemente un entorno gráfico amigable e intuitivo así como la entrada de información para así poder ser analizada por el usuario y luego ser cargada al servicio que realizara el manejo de toda esta información que será el Backend, los archivos de manejo de información serán con formato csv y xml, el Frontend se encargará de tomar esa información y enviarla al Backend en formato json.

El Backend se encargara de tomar toda la información recibida por el Frontend y almacenarla de la manera más adecuada para luego ser devuelta de tal manera que sea posible manipular las listas, visualizar su contenido y reproducir las canciones.

Palabras clave

Python, Frontend, Backend, csv, xml.

Abstract

IPCMusica is an online music player based on the Python programming language, its operation is based on the use of two services the Frontend and Backend, the Frontend is in charge of facilitating the use of the friendly application that implements a graphical environment and Intuitive as well as the information input so that it can be analyzed by the user and then uploaded to the service that will handle all this information, which will be the Backend, the information management files will be in csv and xml format, the Frontend will be It will take care of taking that information and sending it to the Backend in json format.

The Backend will be in charge of taking all the information received by the Frontend and storing it in the most appropriate way to later be returned in such a way that it is possible to manipulate the lists, view their content and play the songs.

Keywords

Python, Frontend, Backend, csv, xml.

Introducción

Python es un lenguaje de programación que en los últimos años a tomado mucha fuerza reemplazando a otros lenguajes de programación debido a sus facilidades y manejo de código, dicho lenguaje será utilizado para la creación de un Backend utilizando flask para los protocolos http y la creación de un Frontend que procesará toda la información que el usuario solicite o envíe al realizar la carga de archivos.

El Backend es una API que se encargará de el manejo de la información de la aplicación, recibirá la información y la almacenará de tal manera que se facilite su uso de igual manera se encargará de devolver esa información para su uso ya sea para ser mostrada y ejecutada por el reproductor o ser exportada a un archivo xml.

El Frontend por otra parte será la vista que tendrá el usuario de la web app con la cual podrá manipularla para lo que necesite, como lo es la carga de archivos, conversión del formato de archivos, reproducción de música, creación de listas y así brindarle al usuario una experiencia agradable he intuitiva que estará formado por el lenguaje Python junto con Django como framework, html para las vistas web y css para los diseños.

Desarrollo del tema

Se desea crear un reproductor de música web basado en el lenguaje de programación Python para el cual se realizará mediante la creación de una API que será el backend y un frontend para la manipulación del usuario.

El backend es un término para referirse al área lógica de una pagina web, es toda la estructura interna que se encargará del funcionamiento correcto de los elementos del sitio, es el área en la que el programador desarrolla la funcionalidad del sitio.

Es de suma importancia ya que sin un backend el sitio web no tendría un motor de funcionamiento ya que éste se encarga de recibir la información, enviada por el usuario mediante el front seguida de una petición que le indicará al backend cual es el objetivo de esta información, tras ser manipulada la información según la solicitud, el backend devolverá una respuesta con la información esperada.

Para su funcionamiento el backend necesita recibir la información de manera externa mediante peticiones http lo que permitiría acceder a los métodos que estén referenciación, realizar el proceso con esa información y devolver una respuesta, pero para poder realizar todo eso es necesaria la implementación de un módulo de Python llamado Flask el cual es un micro framework escrito por Python.



Figura 1. Flask.

Fuente: Flask, 2010.

Pero antes de hablar sobre Flask, ¿Qué es un framework?, un framework en pocas palabras se puede resumir como una plantilla que simplifica la elaboración de una tarea, los framework facilitan la entrega de proyectos en un menor tiempo posible con código más ordenado y limpio con una eficiencia superior, como se mencionaba anteriormente Flask es un mini framework que permite facilitar y agilizar el desarrollo y entrega de proyectos en un menor tiempo posible, sin Flask no sería posible realizar las solicitudes http al backend, para las solicitudes se utilizan algunos métodos request como lo son los mpas comunes el método GET que permite solicitar información, el método POST es utilizado para almacenar información, el método PUT es utilizado para la edición de información y el método DELETE se utiliza para eliminación de información, estos métodos son los utilizados para tomar decisión sobre el funcionamiento de la información recibida.

```
1 from flask import Flask, jsonify, request
2 from flask_cors import CORS
3 app = Flask(__name__)
4 CORS(app)
5 users = []
6
7 @app.route('/usuarios')
8 > def listado(): ...
11
12 @app.route('/usuarios', methods=['POST'])
13 > def addUsuario(): ...
21
22 @app.route('/usuarios/<string:user_name>', methods=['PUT'])
23 > def editUsuario(user_name): ...
37
38 @app.route('/usuarios/<string:user_name>', methods=['DELETE'])
39 > def deleteUsuario(user_name): ...
47
48 if __name__ == '__main__':
49 |     app.run(debug=True, port=4000)
```

Figura 2. Métodos Request

Fuente: Elaboración Propia, 2021.

Para que un Backend sea funcional se requiere de un Frontend que le envíe la información y la reciba procesada de vuelta para dar una respuesta según la

solicitud del usuario, envía una solicitud al Backend mediante los request http antes mencionados, se utilizó Django como framework para el Frontend que como se mencionaba anteriormente el framework facilita el desarrollo de tareas, el Frontend esta compuesto por archivos html que serán las vistas que pueda tener el usuario en cuanto a lo gráfico se refiere, se contará con css para un diseño más agradable a la vista implementado dentro del html, para el funcionamiento lógico de cada visto se utiliza un archivo llamado views que contendrá un método por cada vista el cual cada uno de estos realizara un llamado a las plantillas html y procesará la información recibida o enviada.

La información se obtendrá mediante archivos con formato csv y xml, para los csv será validado el contenido antes de ser convertido a xml para luego ser transformado en json y enviarla al backend y para el xml se cargará y se permitirá editarlo antes de ser enviado.

CSV son las sigrlas de “Comman Separated Values” que significa valores separados por comas, más comúnmente se relacionan los archivos csv con el programa Excel dado que su estructura esta basado en filas y columnas lo cual es la forma en la que se almacena la información en un csv, las columnas quedan definidas por punto y coma (;), mientras que cada fila se define por una línea.

XML es el acrónimo de Extensible Markup Language, es un lenguaje marcado es decir un conjunto de códigos que pueden ser utilizados en análisis de datos o lectura de textos elaborados por computadoras o usuarios.

Básicamente el XML se centra en la simplicidad y facilidad de uso para manejo de información por lo cual se utilizará para la carga masiva a la biblioteca del reproductor.

Conclusiones

Python ofrece múltiples herramientas que le pueden facilitar la vida a las personas tanto para usuario como para desarrolladores.

Actualmente la innovación tecnológica constante obliga a todos por igual a estar actualizados lo que conlleva a manejos de información masiva mediante sitios web y aplicaciones digitales por lo que los archivos XML son de suma importancia para el manejo de tanto contenido.

Para que una app web sea llamativa para el usuario es necesario el desarrollo de un Frontend intuitivo y funcional para que su uso sea de agrado para cualquiera y tenga un éxito ante otros proyectos similares.

El Backend es de suma importancia para cualquier app web ya que al ser el funcionamiento logio del sitio por muy llamativa que sea una web si no es funcional podría provocar la perdida de usuarios.

Referencias bibliográficas

Nestrategia, (2021). *¿Qué es Back End, Front End y Back Office y por qué es importante para tu web?*
<https://nestrategia.com/desarrollo-web-back-end-front-end/>

Ivan de Souza, (2012). *XML: ¿qué es y para qué sirve este lenguaje de marcado?*
<https://rockcontent.com/es/blog/que-es-xml/>

Jose Dominguez Muñoz (2017). *Qué es Flask.*
<https://openwebinars.net/blog/que-es-flask/>