
PROYECTO 3 – Introducción a la programación 2 – Sección: D

201800992 – KEVIN RAUL POZUELOS ESTRADA

Resumen

Aplicación web desarrollada en el lenguaje Python 3 y la utilización de los Frameworks Django en el área del Front-End y Flask en el área del Back-End. La funcionalidad principal de la aplicación web es, la lectura de archivos XML para posteriormente manipular mediante expresiones regulares y crear un archivo de salida (Estadísticas.xml) el cual tiene una composición ordenada y lógica para procesar de manera más fácil la información proporcionada. La aplicación web también proporciona búsqueda de datos mediante dos filtros, “Fecha” y “Código de error” para mostrar de manera grafica los datos conseguidos mediante estas consultas.

Abstract

This Web application was developed in python 3 and the use of Django frameworks in the area of Front-end and the use of Flask for the Back-end, The main use of the web application is the reading of XML files for their later use to manage the regex and output an file (Estadísticas.xml), which has an orderly and logic composition to process the information provided in a easy way.

The web application also provides date search using two filters, "Fecha" (date), and "Código de error" (error code) to show in a graphic way the data obtain through these searches.

Palabras clave

Back-End
Front-end
API
Web
Framework

Keywords

Back-End
Front-end
API
Web
Framework

c. Filtrar: actualización de datos.

d. Reset: eliminación de datos.

Introducción

Proyecto destinado para aprender desarrollo web mediante el lenguaje de Python 3 utilizando frameworks para las áreas del Back-End y Front-End.

Para el área del Back-End se utilizó Flask, en esta parte del proyecto es donde se realizan todas las funcionalidades y API's para luego mandarlas a producción o consumo. En el área del Front-End se utilizó Django aquí es donde se ponen en producción todas las vistas y templates para que la parte visual de la aplicación web se vea atractiva, intuitiva y funcional para cliente. En la parte del Front-End es donde se hace posible la conexión o consumo de las API's realizadas en el área del Back-End para que sea funcional la parte del Front-End

Desarrollo del tema

Aplicación web desarrollada en Python 3 conformada por dos partes, Back-End y Front-End. Para procesar archivos XML mediante expresiones regulares para realizar un archivo de salida en formato XML con mas etiquetas que describen de manera mas optima los eventos, fechas, errores, reportadores y afectados mediante el consumo de diferentes API's y métodos HTTP.

Las funcionalidades principales son:

- a. Enviar: guardar, crear, procesar datos.
- b. Petición: lectura, parseo de datos

WEB API

Una API es una interfaz de programación de aplicaciones (del inglés API: Application Programming Interface). Es un conjunto de rutinas que provee acceso a funciones de un determinado software.

Son publicadas por los constructores de software para permitir acceso a características de bajo nivel o propietarias, detallando solamente la forma en que cada rutina debe ser llevada a cabo y la funcionalidad que brinda, sin otorgar información acerca de cómo se lleva a cabo la tarea. Son utilizadas por los programadores para construir sus aplicaciones sin necesidad de volver a programar funciones ya hechas por otros, reutilizando código que se sabe que está probado y que funciona correctamente.

¿Qué es una API Rest?

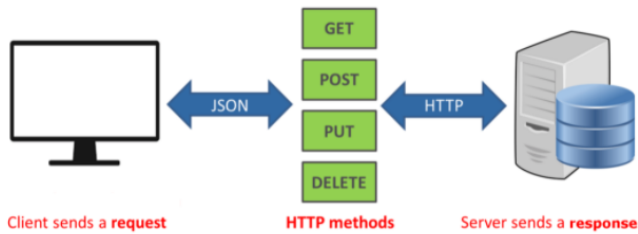
Antes de introducir el concepto de Rest API, es necesario explicar qué es una API (Interfaz de programación de aplicaciones). Es un conjunto de requisiciones que permite la comunicación de datos entre aplicaciones.

Para eso, la API utiliza requisiciones HTTP responsables de las operaciones básicas necesarias para la manipulación de datos.

Las principales solicitudes son:

- POST: crea datos en el servidor;
- GET: lectura de datos en el host;

- DELETE: borra la información;
- PUT: registro de actualizaciones.



Rest, que es la abreviación de Representational State Transfer, es un conjunto de restricciones que se utilizan para que las solicitudes HTTP cumplan con las directrices definidas en la arquitectura.

Básicamente, las restricciones determinadas por la arquitectura Rest son:

- Cliente-servidor: las aplicaciones existentes en el servidor y el cliente deben estar separadas.
- Sin estado: las requisiciones se realizan de forma independiente, es decir, cada una ejecuta solo una determinada acción
- Caché: la API debe utilizar la caché para evitar llamadas recurrentes al servidor.
- Interfaz uniforme: agrupa otros cuatro conceptos en los que se determina que los recursos deben ser identificados, la manipulación de los recursos debe ser a través de la representación, con mensajes autodescriptivos y utilizando enlaces para navegar por la aplicación.

¿Qué tipos de API existen?

Básicamente, existen tres tipos de APIs:

- **Privados:** que se utilizan internamente entre las aplicaciones de una empresa, es decir, localmente.
- **Socios:** que se utilizan entre socios comerciales o para permitir la integración entre diferentes softwares.
- **Público:** que se puede utilizar libremente. A menudo, la empresa los pone a disposición para que los desarrolladores puedan integrarse con otras aplicaciones.

¿Para qué sirve API Rest?

Existe una gran variación en las formas en que se utilizan los API. Las redes sociales, por ejemplo, proporcionan un API que se puede utilizar en otros sitios para recuperar información de una página. Existen varios plugins de WordPress que acceden a las redes sociales a través de ellos y transforman el resultado de esta interacción en pequeñas visualizaciones del estado actual de la página correspondiente.

Así, si un usuario quiere darle like a la página, por ejemplo, no es necesario que abandone el sitio original para esta acción.

Al hacer clic en el botón Me Gusta, hay una llamada al API para completar esta operación. Para que eso sea posible, las redes sociales proporcionan un token con la debida autorización para que API tenga acceso a la información.

Al igual que las redes sociales, los APIs también se utilizan en los sitios de ecommerce para acceder a los intermediarios de pago y completar las transacciones de compra.

Por lo tanto, el API sirve para comunicarse entre aplicaciones para intercambiar informaciones de forma rápida y segura.

¿Qué importancia tiene la API Rest para el sitio web de la empresa?

Uno de los factores más importantes al utilizar un sitio web con Rest API es la facilidad de comunicación con otras aplicaciones. Las interfaces permiten agregar funcionalidades o información al sitio web de una manera simple, rápida y segura.

De esa forma, es posible crear sitios web o aplicaciones en la nube para acceder a datos de sistemas más robustos, como ERPs, sea cual sea la plataforma desde la que se originan los datos. Esa función permite a las empresas acceder a informaciones actualizadas mucho más rápidamente.

¿Cuáles son las ventajas de utilizar una API Rest?

Las características de la arquitectura Rest API brindan una serie de beneficios a las aplicaciones que utilizan este concepto.

Conclusiones

El proyecto nos sirve para comprender el funcionamiento de las aplicaciones web.

Los frameworks son esenciales en las tecnologías actuales

Las aplicaciones web están conformadas por dos partes Backend y FrontEnd y se encargan de la funcionalidad de la aplicación web.

Es importante aprender a realizar aplicaciones web ya que es una tecnología que en el mercado laboral actual es necesario y muy demandado.

Referencias bibliográficas

CHALLENGER-PÉREZ, Ivet; DÍAZ-RICARDO, Yanet; BECERRA-GARCÍA, Roberto Antonio. El lenguaje de programación Python. *Ciencias Holguín*, 2014, vol. 20, no 2, p. 1-13.

GONZÁLEZ DUQUE, Raúl. Python para todos. 2014.

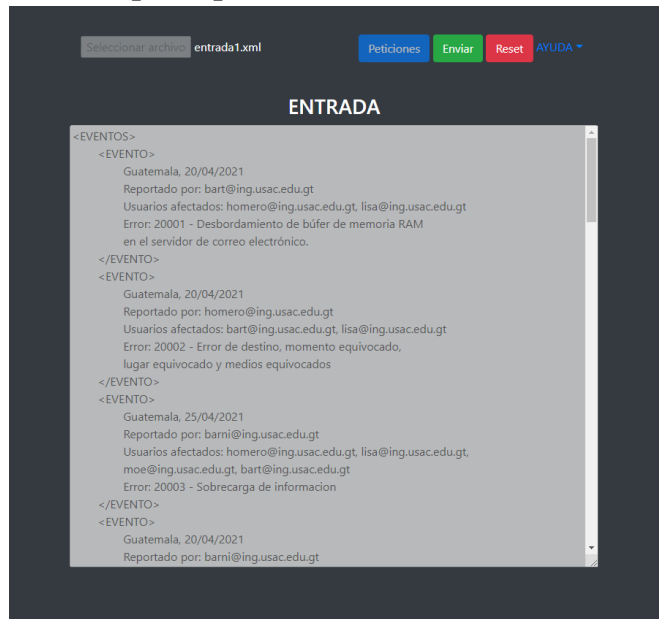
FORCIER, Jeff; BISSEX, Paul; CHUN, Wesley J. *Python web development with Django*. Addison-Wesley Professional, 2008.

GRINBERG, Miguel. *Flask web development: developing web applications with python*. "O'Reilly Media, Inc.", 2018.

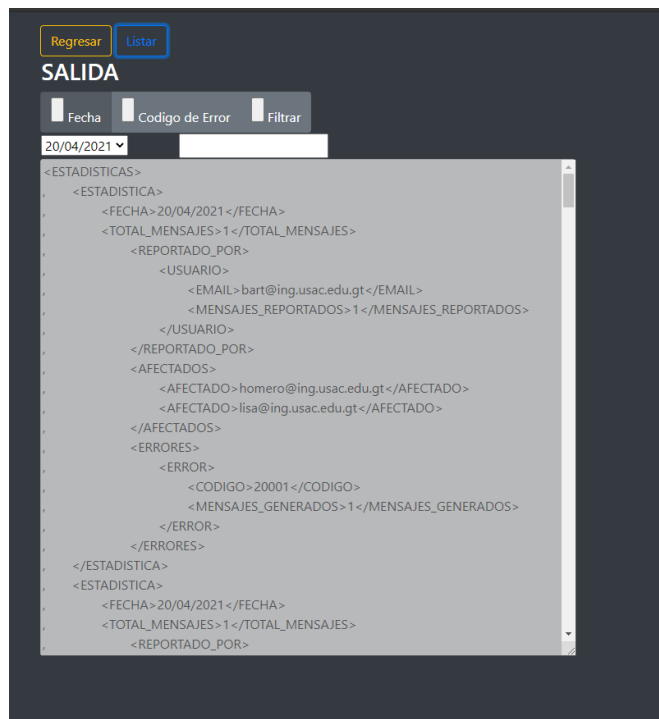
HUMPHREY, Arthur. Shake flask to fermentor: what have we learned?. *Biotechnology Progress*, 1998, vol. 14, no 1, p. 3-7.

ANEXOS

Pantalla principal



Pantalla de Peticiones



DATOS PERSONALES

